[MS-PPT]:

PowerPoint (.ppt) Binary File Format

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation ("this documentation") for protocols, file formats, data portability, computer languages, and standards support. Additionally, overview documents cover inter-protocol relationships and interactions.
- **Copyrights**. This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you can make copies of it in order to develop implementations of the technologies that are described in this documentation and can distribute portions of it in your implementations that use these technologies or in your documentation as necessary to properly document the implementation. You can also distribute in your implementation, with or without modification, any schemas, IDLs, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications documentation.
- **No Trade Secrets**. Microsoft does not claim any trade secret rights in this documentation.
- **Patents**. Microsoft has patents that might cover your implementations of the technologies described in the Open Specifications documentation. Neither this notice nor Microsoft's delivery of this documentation grants any licenses under those patents or any other Microsoft patents. However, a given Open Specifications document might be covered by the Microsoft Open Specifications Promise or the Microsoft Community Promise. If you would prefer a written license, or if the technologies described in this documentation are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.
- **Trademarks**. The names of companies and products contained in this documentation might be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit www.microsoft.com/trademarks.
- **Fictitious Names**. The example companies, organizations, products, domain names, email addresses, logos, people, places, and events that are depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

Reservation of Rights. All other rights are reserved, and this notice does not grant any rights other than as specifically described above, whether by implication, estoppel, or otherwise.

Tools. The Open Specifications documentation does not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments, you are free to take advantage of them. Certain Open Specifications documents are intended for use in conjunction with publicly available standards specifications and network programming art and, as such, assume that the reader either is familiar with the aforementioned material or has immediate access to it.

Revision Summary

Date	Revision History	Revision Class	Comments
6/27/2008	1.0	New	First release
8/15/2008	1.01	Minor	Revised and edited the technical content
1/16/2009	1.02	Minor	Updated the Intellectual Property Rights Notice
7/13/2009	1.03	Major	Changes made for template compliance
8/28/2009	1.04	Editorial	Revised and edited the technical content
11/6/2009	1.05	Editorial	Revised and edited the technical content
2/19/2010	2.0	Minor	Updated the technical content
3/31/2010	2.01	Editorial	Revised and edited the technical content
4/30/2010	2.02	Editorial	Revised and edited the technical content
6/7/2010	2.03	Editorial	Revised and edited the technical content
6/29/2010	2.04	Editorial	Changed language and formatting in the technical content.
7/23/2010	2.05	Major	Significantly changed the technical content.
9/27/2010	2.05	None	No changes to the meaning, language, or formatting of the technical content.
11/15/2010	2.05	None	No changes to the meaning, language, or formatting of the technical content.
12/17/2010	2.05	None	No changes to the meaning, language, or formatting of the technical content.
3/18/2011	2.05	None	No changes to the meaning, language, or formatting of the technical content.
6/10/2011	2.05	None	No changes to the meaning, language, or formatting of the technical content.
1/20/2012	3.0	Major	Significantly changed the technical content.
4/11/2012	3.0	None	No changes to the meaning, language, or formatting of the technical content.
7/16/2012	3.0	None	No changes to the meaning, language, or formatting of the technical content.
10/8/2012	3.0	None	No changes to the meaning, language, or formatting of the technical content.
2/11/2013	3.0	None	No changes to the meaning, language, or formatting of the technical content.
7/30/2013	3.0	None	No changes to the meaning, language, or formatting of the technical content.
11/18/2013	3.1	Minor	Clarified the meaning of the technical content.
2/10/2014	3.1	None	No changes to the meaning, language, or formatting of the technical content.

Date	Revision History	Revision Class	Comments
4/30/2014	3.1	None	No changes to the meaning, language, or formatting of the technical content.
7/31/2014	3.1	None	No changes to the meaning, language, or formatting of the technical content.
10/30/2014	3.1	None	No changes to the meaning, language, or formatting of the technical content.
3/16/2015	4.0	Major	Significantly changed the technical content.
9/4/2015	4.0	None	No changes to the meaning, language, or formatting of the technical content.
7/15/2016	4.0	None	No changes to the meaning, language, or formatting of the technical content.

Table of Contents

1 Introduction

This document specifies the binary file format for a PowerPoint (PPT) file (.ppt) used by Microsoft PowerPoint 97, Microsoft PowerPoint 2000, Microsoft PowerPoint 2002, and Microsoft Office PowerPoint 2003. A PPT file is a collection of records and structures that specify **slides**, **shapes**, pictures, audio, video, text, and other **presentation** content. This content can then be delivered to an audience by means of a **slide show**.

Each record has a common header that specifies the record type and any additional data that follows. This file format provides an efficient way to parse only records that contain content of interest to a particular implementation and to skip any other records.

Sections 1.7 and 2 of this specification are normative. All other sections and examples in this specification are informative.

1.1 Glossary

This document uses the following terms:

ActiveX control: A reusable software control, such as a check box or button, that uses ActiveX technology and provides options to users or runs macros or scripts that automate a task. See also ActiveX object.

add-in: Supplemental functionality that is provided by an external application or macro to extend the capabilities of an application.

atom record: A record that contains presentation data. Analogous to a file system, atom records are similar to files that contain data and container records are similar to directories that provide structure and hierarchy for atom records.

Audio Interchange File Format (AIFF): A sound file format that was originally used on Macintosh and Silicon Graphics (SGI) computers. AIFF stores waveform files in an 8-bit monaural format. See also **waveform (WAV)**.

Audio Video Interleaved (AVI): A multimedia file format for sound and video that uses the Microsoft Resource Interchange File Format (RIFF).

Augmented Backus-Naur Form (ABNF): A modified version of Backus-Naur Form (BNF), commonly used by Internet specifications. ABNF notation balances compactness and simplicity with reasonable representational power. ABNF differs from standard BNF in its definitions and uses of naming rules, repetition, alternatives, order-independence, and value ranges. For more information, see [RFC5234].

big-endian: Multiple-byte values that are byte-ordered with the most significant byte stored in the memory location with the lowest address.

bookmark: An entity that is used in a document to denote the beginning and ending character positions of specific text in the document, and optionally, metadata about that text or its relationship to other referenced parts of the document.

Boolean: An operation or expression that can be evaluated only as either true or false.

build: An animation effect that is applied to content on a **presentation slide**. The complete animation for a given piece of content is specified as a sequence of builds, where each build causes parts of the content to appear, disappear, move, or be emphasized in some manner.

build identifier: An integer that identifies a **build**.

code page: An ordered set of characters of a specific script in which a numerical index (code-point value) is associated with each character. Code pages are a means of providing support for character sets and keyboard layouts used in different countries. Devices such as the display and keyboard can be configured to use a specific code page and to switch from one code page (such as the United States) to another (such as Portugal) at the user's request.

color scheme: A table of color values that enables colors to be referenced by an index value in the table instead of a color value. See also color palette.

color space: A system that describes color numerically by mapping color components to a multidimensional coordinate system. The number of dimensions is typically two, three, or four. For example, if colors are expressed as a combination of the three components red, green, and blue, a three-dimensional space can describe all possible colors. Grayscale colors can be mapped to a two-dimensional color space. If transparency is considered a component, four dimensions are appropriate. Also referred to as color model.

computer name: The DNS or NetBIOS name.

container record: A record that defines the structure and hierarchy of **atom records** and other container records.

cyclic redundancy check (CRC): An algorithm used to produce a checksum (a small, fixed number of bits) against a block of data, such as a packet of network traffic or a block of a computer file. The CRC is a broad class of functions used to detect errors after transmission or storage. A CRC is designed to catch random errors, as opposed to intentional errors. If errors might be introduced by a motivated and intelligent adversary, a cryptographic hash function should be used instead.

encrypted document: A document that was converted from plaintext into cipher text to disguise the content of the document when it is stored or sent.

envelope: A container that stores the information that is used to send a document, workbook, or presentation in an email message. The information includes the intended recipients, the subject of the email message, and any attachments to be included.

external object: An object such as a movie file, audio file, OLE object, or hyperlink that can be associated with a presentation or document.

floating-point number: A number that is represented by a mantissa and an exponent according to a given base. The mantissa is typically a value between "0" and "1". To find the value of a floating-point number, the base is raised to the power of the exponent, and the mantissa is multiplied by the result.

footer: One or more lines of text in the bottom margin area of a page in a document or a slide in a presentation. A footer typically contains elements such as the page number and the name of the file.

grid unit: A unit of linear measurement that is equal to 1/1024 master unit or 1/589824 inch.

guide: A vertical or a horizontal line that can be set as part of the user interface to position and align content on a slide.

handout master slide: A slide that defines layout and positioning information for handout pages, which are pages that are optimized for printing a presentation.

handout slide: A slide that is taken from a presentation and then printed and distributed to the audience of the presentation. Users can select the number of slides to be printed on each page in a handout.

header: A line, or lines, of content in the top margin area of a page in a document or a slide in a presentation. A header typically contains elements such as the title of the chapter, the title of the document, a page number, or the name of the author.

hue-saturation-luminance (HSL): A color model that defines a color by using three dimensions: hue, the color itself; saturation, the purity of the color; and luminance, the amount of light that is either reflected or absorbed by the color. See also **color scheme** and **color space**.

hyperlink: A relationship between two anchors, as described in [RFC1866].

kiosk mode: A display mode in which a single window expands to fill the whole screen. In kiosk mode, the toolbar and menu bar are not displayed, and the desktop is inaccessible.

little-endian: Multiple-byte values that are byte-ordered with the least significant byte stored in the memory location with the lowest address.

macro: A set of instructions that are recorded or written, and then typically saved to a file. When a macro is run, all of the instructions are performed automatically.

main master slide: A slide that defines the formatting and content that can be used by presentation slides. If a slide uses formatting and content from a main master slide, it is referred to as following a main master slide.

master unit: A unit of linear measurement that is equal to 1/576 inch.

Media Control Interface (MCI): A part of the Windows API that enables an application to control multimedia devices. The term is also used to generically describe any media object that can be played through the interface.

metafile: A file that stores an image as graphical objects, such as lines, circles, and polygons, instead of pixels. A metafile preserves an image more accurately than pixels when an image is resized.

Musical Instrument Digital Interface (MIDI): A specification of the MIDI Manufacturers Association (MMA). The specification for Musical Instrument Digital Interface (MIDI) defines a protocol for describing music data, such as note on and note off messages; a file format for storing music data, called Standard MIDI; and a standard hardware interface.

named show: A named sequence of slides that can be displayed as a slide show.

notes master slide: A slide that defines the formatting and content that can be used by notes slides for a presentation. If a notes slide uses formatting and content from a notes master slide, it is referred to as following a notes master slide.

notes slide: A slide that contains presentation notes or other information that is not displayed during a slide show. The formatting and content of a notes slide can derive from a notes master slide.

Object Linking and Embedding (OLE): A technology for transferring and sharing information between applications by inserting a file or part of a file into a compound document. The inserted file can be either embedded or linked. See also embedded object and linked object.

OLE compound file: A form of structured storage, as described in [MS-CFB]. A compound file allows independent storages and streams to exist within a single file.

OLE object: An object that supports the **Object Linking and Embedding (OLE)** protocol.

OLE verb: An action defined by an OLE-linked object that specifies what behaviors can be applied to it.

persist object: A top-level object that can be independently persisted and that forms the basis of an incremental save model. A persist object is one of the following: presentation document, main master slide, title master slide, handout master slide, notes master slide, presentation slide, notes slide, OLE object storage, or Microsoft Visual Basic for Applications (VBA) project storage.

persist object directory: A table of persist object identifiers and stream offsets to where persist objects can be found. Each user edit stores a persist object directory that identifies where any new and modified persist objects can be found.

persist object identifier: A unique identifier that is associated with a persist object and is stored in a persist object directory.

placeholder: A character or symbol that is used in place of an actual value, text, or object. The actual value that the placeholder represents is unknown or unavailable at the current time, or is not displayed for security reasons.

placeholder shape: A special type of shape in a presentation that usually includes common visual properties. These are used to effect a uniform look among different slides, or to uniformly represent meta-information about each slide.

point: A unit of measurement for fonts and spacing. A point is equal to 1/72 of an inch.

Portable Network Graphics (PNG): A bitmap graphics file format that uses lossless data compression and supports variable transparency of images (alpha channels) and control of image brightness on different computers (gamma correction). PNG-format files have a .png file name extension.

presentation: A collection of slides that are intended to be viewed by an audience.

presentation broadcast: A feature that enables users to run a presentation over the web. The presentation is saved in HTML format and can contain audio and video. It can also be recorded and saved for viewing later.

presentation comment: A text note that is attached to a slide to enable readers of a presentation to provide feedback to the presentation author.

presentation slide: A slide that contains the content that can be displayed during a slide show. A presentation slide can derive formatting and content from a main master slide or a title master slide.

programmable tag: A name/value pair. The value can be either a Unicode string or binary data.

red-green-blue (RGB): A color model that describes color information in terms of the red (R), green (G), and blue (B) intensities in a color.

routing slip: Information that specifies how a document is to be distributed from a document originator and processed by one or more recipients. It also specifies subject and message body text that is associated with the document routing process and routing status or workflow information.

seed: A value that is greater than or equal to all other values in a set of values and is used to create the next value in the set.

shape: A collection of qualifiers, such as names, and quantifiers, such as coordinates, that is used to represent a geometric object. A shape can be contained in a document, file structure, run-time structure, or other medium.

shape identifier: An integer that corresponds to a shape object or an instantiation of a shape object.

slide: A frame that contains text, shapes, pictures, or other content. A slide is a digital equivalent to a traditional film slide.

slide layout: An organizational scheme, such as Title Only or Comparison, for content on a presentation slide.

slide show: A delivery of a sequence of presentation slides, typically to an audience.

smart tag: A feature that adds the ability to recognize and label specific data types, such as people's names, within a document and displays an action button that enables users to perform common tasks for that data type.

sound identifier: A unique identifier for an embedded or linked sound.

table object: A group of shapes that are arranged in rows and columns to form a table.

text ruler: A collection of settings for tabs, margins, and indentation of text. See also ruler.

time condition: A logical condition that can be evaluated, or an event that can be triggered, to determine whether timed object behavior starts or ends. Conditions include items such as the start or end of time nodes, keyboard presses, mouse clicks, or delegate events. See also **time node**.

time node: A record or parent node that stores the information that is necessary to cause a time- or action-based effect to occur. Each time node has a corresponding object to which an effect is applied. It can be used randomly, simultaneously, or sequentially, and it can be used to specify certain time-based effects between objects that are being animated. Effects include visual and media behaviors.

title master slide: A slide that defines the formatting and content that can be used by presentation slides that have a title slide layout. If a slide uses formatting and content from a title master slide, it is referred to as following a title master slide.

Unicode: A character encoding standard developed by the Unicode Consortium that represents almost all of the written languages of the world. The **Unicode** standard [UNICODE5.0.0/2007] provides three forms (UTF-8, UTF-16, and UTF-32) and seven schemes (UTF-8, UTF-16, UTF-16 BE, UTF-16 LE, UTF-32, UTF-32 LE, and UTF-32 BE).

Uniform Resource Locator (URL): A string of characters in a standardized format that identifies a document or resource on the World Wide Web. The format is as specified in [RFC1738].

Universal Naming Convention (UNC): A string format that specifies the location of a resource. For more information, see [MS-DTYP] section 2.2.57.

user edit: A set of persist objects and a persist object directory that represent changes made by a user. Each time a file is written, a user edit that contains only those persist objects that are new or modified from the previous user edit can be appended to the pre-existing record stream. The persist objects and the persist object directory in a later user edit supersede those in a previous user edit.

VBA project: A collection of the modules, class modules, and user forms that are needed to create an application. Modules, class modules, and user forms can be imported into and exported from a project.

Vector Markup Language (VML): A system of marking up or tagging two-dimensional vector graphics for publication on the World Wide Web. VML graphics are scalable and editable, and typically require less disk space and less time to download.

Visual Basic for Applications (VBA): A macro-based programming language that derives from Microsoft Visual Basic and can be used to customize and extend an application. Unlike Visual Basic, VBA code and macros can be run only from within a host application that supports VBA.

waveform (WAV): A file format in which Windows stores sounds as waveforms. Depending on the sampling frequency, whether the sound is monaural or stereo, and whether 8 or 16 bits are used for each sample, one minute of sound can occupy as little as 644 kilobytes or as much as 27 megabytes of storage. Waveform files have a .wav file name extension.

MAY, SHOULD, MUST, SHOULD NOT, MUST NOT: These terms (in all caps) are used as defined in [RFC2119]. All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

1.2 References

Links to a document in the Microsoft Open Specifications library point to the correct section in the most recently published version of the referenced document. However, because individual documents in the library are not updated at the same time, the section numbers in the documents may not match. You can confirm the correct section numbering by checking the Errata.

1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We will assist you in finding the relevant information.

[ECMA-376] ECMA International, "Office Open XML File Formats", 1st Edition, ECMA-376, December 2006, http://www.ecma-international.org/publications/standards/Ecma-376.htm

[Embed-Open-Type-Format] Nelson, P., "Embedded OpenType (EOT) File Format", W3C Member Submission, March 2008, http://www.w3.org/Submission/2008/SUBM-EOT-20080305/

[IEC-RGB] International Electrotechnical Commission, "Colour Measurement and Management in Multimedia Systems and Equipment - Part 2-1: Default RGB Colour Space - sRGB", May 1998, http://webstore.iec.ch/webstore/webstore.nsf/artnum/025408

[JFIF] Hamilton, E., "JPEG File Interchange Format, Version 1.02", September 1992, http://www.w3.org/Graphics/JPEG/jfif.txt

[MC-LOGFONT] Microsoft Corporation, "Windows GDI LOGFONT", http://msdn.microsoft.com/en-us/library/dd145037.aspx

[MS-CFB] Microsoft Corporation, "Compound File Binary File Format".

[MS-DTYP] Microsoft Corporation, "Windows Data Types".

[MS-EMF] Microsoft Corporation, "Enhanced Metafile Format".

[MS-LCID] Microsoft Corporation, "Windows Language Code Identifier (LCID) Reference".

[MS-ODRAW] Microsoft Corporation, "Office Drawing Binary File Format".

[MS-OFFCRYPTO] Microsoft Corporation, "Office Document Cryptography Structure".

[MS-OFORMS] Microsoft Corporation, "Office Forms Binary File Formats".

[MS-OLEPS] Microsoft Corporation, "Object Linking and Embedding (OLE) Property Set Data Structures".

[MS-OSHARED] Microsoft Corporation, "Office Common Data Types and Objects Structures".

[MS-OVBA] Microsoft Corporation, "Office VBA File Format Structure".

[MS-WMF] Microsoft Corporation, "Windows Metafile Format".

[MSFT-RTF] Microsoft Corporation, "Rich Text Format (RTF) Specification", version 1.9.1, March 2008, http://www.microsoft.com/en-us/download/details.aspx?id=10725

[RFC1950] Deutsch, P., and Gailly, J-L., "ZLIB Compressed Data Format Specification version 3.3", RFC 1950, May 1996, http://www.ietf.org/rfc/rfc1950.txt

[RFC1951] Deutsch, P., "DEFLATE Compressed Data Format Specification version 1.3", RFC 1951, May 1996, http://www.ietf.org/rfc/rfc1951.txt

[RFC2083] Boutell, T., et al., "PNG (Portable Network Graphics) Specification Version 1.0", RFC 2083, March 1997, http://www.ietf.org/rfc/rfc2083.txt

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, http://www.rfc-editor.org/rfc/rfc2119.txt

[RFC2781] Hoffman, P., and Yergeau, F., "UTF-16, an encoding of ISO 10646", RFC 2781, February 2000, http://www.rfc-editor.org/rfc/rfc2781.txt

[RFC3629] Yergeau, F., "UTF-8, A Transformation Format of ISO 10646", STD 63, RFC 3629, November 2003, http://www.ietf.org/rfc/rfc3629.txt

[RFC3986] Berners-Lee, T., Fielding, R., and Masinter, L., "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, January 2005, http://www.ietf.org/rfc/rfc3986.txt

[RFC5234] Crocker, D., Ed., and Overell, P., "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008, http://www.rfc-editor.org/rfc/rfc5234.txt

[XML] World Wide Web Consortium, "Extensible Markup Language (XML) 1.0 (Fourth Edition)", W3C Recommendation 16 August 2006, edited in place 29 September 2006, http://www.w3.org/TR/2006/REC-xml-20060816/

1.2.2 Informative References

[ASF] Microsoft Corporation, "Advanced Systems Format Specification", December 2004, http://download.microsoft.com/download/7/9/0/790fecaa-f64a-4a5e-a430-0bccdab3f1b4/ASF_Specification.doc

[ISO/IEC29500-1:2012] ISO/IEC, "Information Technology -- Document description and processing languages -- Office Open XML File Formats -- Part 1: Fundamentals and Markup Language Reference", ISO/IEC 29500-1:2012, http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=61750

[MS-OLEDS] Microsoft Corporation, "Object Linking and Embedding (OLE) Data Structures".

[MSDN-COM] Microsoft Corporation, "Component Object Model", http://msdn.microsoft.com/en-us/library/aa286559.aspx

[MSDN-CP] Microsoft Corporation, "Code Page Identifiers", http://msdn.microsoft.com/en-us/library/dd317756(VS.85).aspx

[MSDN-FILE] Microsoft Corporation, "Naming Files, Paths, and Namespaces", http://msdn.microsoft.com/en-us/library/aa365247.aspx

[MSFT-UMWNSNS] Microsoft Corporation, "Using MS Windows NT Server NetShow Services", http://technet.microsoft.com/en-us/library/bb676131.aspx

1.3 Structure Overview (Synopsis)

1.3.1 Presentation Document

A presentation is a collection of **presentation slides** intended to be viewed by an audience during a slide show. In addition to presentation slides and slide show settings, it stores **notes slides** containing speaker notes and a **handout master slide** that determines how to print presentation handouts.

The following sections elaborate on the various types of slides, their contents, and slide show behaviors.

See the Document Types (section 2.4) for records pertaining to the document as a whole.

1.3.2 Slides

Slides are the basic building blocks of a presentation. Each slide contains user data such as shapes, text, animations, and media. A slide show sequentially displays each presentation slide within the presentation to an audience.

The **main master slides** and **title master slides** are special types of slides that define common visual properties such as content and formatting for a group of presentation slides. A presentation slide inherits visual properties defined in a main master slide or title master slide. Inheritance of visual properties is referred to as following that main master slide or title master slide. Likewise, the **notes master slide** and the handout master slide serve a similar purpose and provide common visual properties for all notes slides and all printed handouts respectively.

See the Slide Types (section 2.5) for records pertaining to slides. Also see the Slides Example (section 3.5) for examples about slides.

1.3.3 Shapes

Shapes are the primary way to represent data on a slide. Different types of shapes such as **placeholder shapes**, pictures, and graphs allow users to add a variety of content to a slide. Shapes on a master slide define common data for groups of shapes.

See the Shape Types (section 2.7) for records pertaining to shapes. Also see Office Drawing Binary File Format Structure Specification, [MS-ODRAW], for more information about shapes.

1.3.4 Placeholders Shapes

Placeholder shapes are specialized shapes that serve as containers for a variety of objects. Title and body placeholder shapes can be used to extract an outline text representation of the document. Header, footer, date, and slide-number placeholder shapes can be used to uniformly present meta-information on each slide. Various object placeholder shapes can be used to provide clues to insert specific types of shapes, such as tables or charts.

When contained within presentation slides or notes slides, a placeholder shape inherits its visual properties from a corresponding placeholder shape on a main master slide, title master slide, or notes master slide.

1.3.5 External Objects

Slides can contain objects that link to resources external to the presentation document. Presenters can activate linked objects to access external resources during a slide show. Examples of **external objects** are embedded and linked audio, linked video, embedded and linked **OLE objects**, and hyperlinks.

See the External Object Types (section 2.10) for records pertaining to external objects.

1.3.6 Animation

One or more animation effects can be applied to a shape or the text in a shape. Animation effects vary one or more of the shape's properties over a defined period of time during a slide show. Each slide has a timeline that stores timing and sequencing information for all animations on that slide. Animations are displayed according to the timeline during a slide show.

1.3.6.1 Timeline

A slide timeline defines the order, duration, start conditions, and exit conditions for all animations on a slide. Each slide has exactly one timeline.

Time nodes are the basic building blocks for timelines. Each time node stores the order, duration, start conditions, and exit conditions for an animation effect. A timeline can contain an unlimited number of time nodes organized in a tree structure.

There are two main types of time nodes, as described in the following table:

Time node	Description
Parallel	A time node whose child nodes are executed in parallel.
Sequence	A time node whose child nodes are executed only after the previous sibling has started to be executed.

1.3.6.2 Conditional Properties

Conditional properties allow more granular control and choreography between time nodes in a timeline. These properties specify conditions that need to be met before a time node starts or finishes executing. For example, it allows an animation to be started when a slide is first displayed or when triggered by a user action.

There are four such conditional properties, as described in the following table:

Time node	Description
Start Condition	Conditions that control when a time node starts.
Previous Condition	Conditions that control when the timeline goes back to the previous time node.
Next Conditions	Conditions that control when the timeline advances to next time node.
End Conditions	Conditions that need to be met for a time node to end.

1.3.6.3 Behaviors

Animation effects vary one or more properties of a shape. Variation of a property over time is called a behavior. Available behaviors are dependent on the type of shape being animated. For example, a geometric shape supports behaviors that change its color and size while a video shape supports behaviors that define its playback.

Behaviors belong to the basic types described in the following table:

Behavior type	Description
Animate	Animates any property of an object that is either a numerical or a string value.
Animate Color	Animates the color values of an object.
Animate Effect	Allows image transformations and filter animations on an object.
Animate Motion	Animates position properties of an object by using either key-frame data or detailed path descriptions that include Bezier curves or lines.
Animate Rotation	Animates the orientation of an object.
Animate Scale	Animates the width and height of an object over time.

A time node can combine multiple animations to create complex effects. For example, the "flash bulb" animation, which scales a shape larger while simultaneously fading it, uses two animation behavior elements. An example is shown in the following figure:

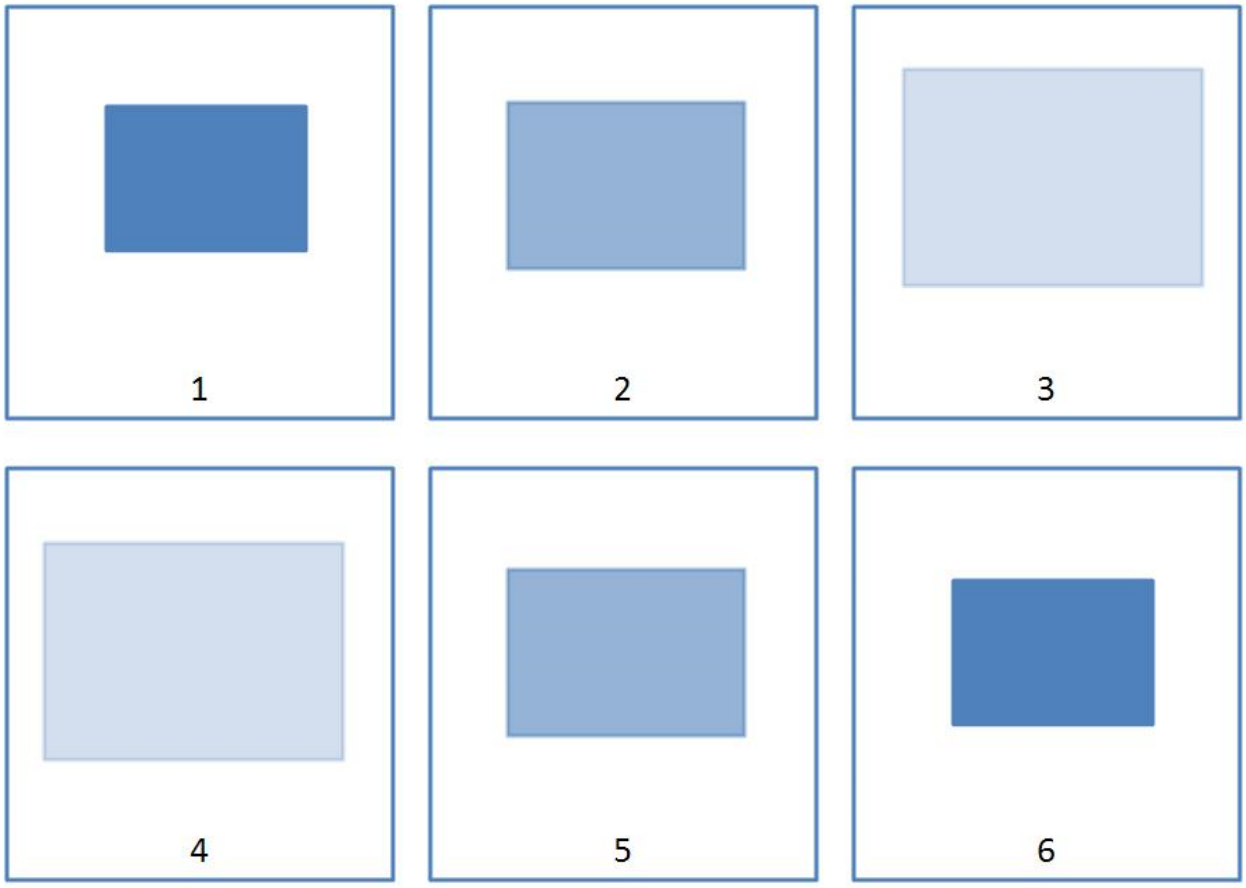


Figure 1: Simultaneous color and scale animations

1.3.6.4 Builds

Some shapes, such as text, diagrams, and graphs, have component objects. A text placeholder shape, for example, can contain multiple lines of text. A pie chart can contain multiple slices. Build order allows for the animation of component objects within a shape.

An example of animating a pie chart by category is shown in the following figure:

Figure 2: A chart-build animation

See the Animation Types (section 2.8) for records pertaining to animation.

1.3.7 Slide show

A slide show delivers presentation slides to an audience. During a slide show, each presentation slide is displayed sequentially in the order it is stored. Users can add animation effects between slides, and they can create a custom sequence of slides by using a **named show**.

1.3.7.1 Named Show

By default, a slide show sequentially displays all presentation slides. A named show specifies a sequence of presentation slides during a slide show that is different from the order in which the presentation slides themselves are stored. Named shows can contain a subset of all presentation slides in a document.

1.3.7.2 Slide Transitions

Slide transitions are the animation effects displayed between presentation slides. An example of a slide that has a "push" slide transition is shown in the following figure:

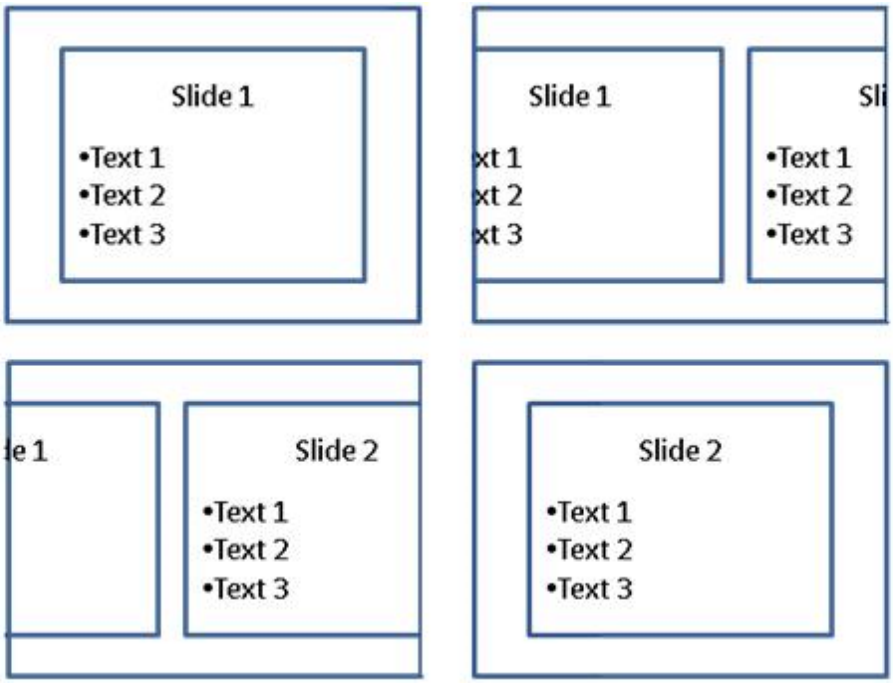


Figure 3: A "push" slide transition

See the Slide Show Types (section 1.3.7) for records pertaining to slide show.

1.3.8 Byte Ordering

Some computer architectures number bytes in a binary word from left to right, a format that is referred to as **big-endian**. The packet diagrams for this documentation are big-endian. Other architectures number the bytes in a binary word from right to left, a format that is referred to as **little-endian**. The underlying file format enumerations, objects, and records are little-endian.

Using big-endian and little-endian methods, the number 0x12345678 is stored as shown in the following table:

Byte order	Byte 0	Byte 1	Byte 2	Byte 3
Big-endian	0x12	0x34	0x56	0x78
Little-endian	0x78	0x56	0x34	0x12

Unless otherwise specified, all data in the PowerPoint Binary File Format is stored in little-endian format.

1.4 Relationship to Protocols and Other Structures

This file format is an **OLE compound file** as described in [MS-CFB]. It is dependent on the structures described in the following references:

- [MS-ODRAW] for the persistence format for shapes.
- [MS-OVBA] for the persistence format for a **VBA project**.
- [MS-OFFCRYPTO] for the persistence format for document signing, information rights management, document encryption and obfuscation.
- [MS-OSHARED] for the persistence format for additional common structures.

This file format has been superseded by [ECMA-376] in Microsoft Office PowerPoint 2007 and by [ISO/IEC29500-1:2012] in Microsoft PowerPoint 2010 and Microsoft PowerPoint 2013.

1.5 Applicability Statement

This document specifies a persistence format for presentation content and templates, which can include slides, drawing objects, text, images, transitions, and animations. This persistence format is applicable when the primary presentation format for the contained information is electronic.

This persistence format is applicable for use as a stand-alone document, and for containment within other documents as an embedded object as described in [MS-OLEDS].

This persistence format provides interoperability with applications that create or read documents conforming to this structure, including PowerPoint 97, PowerPoint 2000, PowerPoint 2002, and Office PowerPoint 2003. This persistence format can also be used for interoperability with Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 when compatibility with PowerPoint 97, PowerPoint 2000, PowerPoint 2002, and Office PowerPoint 2003 is a primary concern.

1.6 Versioning and Localization

None.

1.7 Vendor-Extensible Fields

This persistence format can be extended by storing information in streams and storages that are not described in section 2. Implementations are not required to preserve or remove additional streams or storages when modifying an existing document.

2 Structures

2.1 File Streams and Storages

As an OLE compound file, this file format specification is organized as a hierarchy of storages and streams as specified in [MS-CFB]. The following sections list the top-level storages and streams found in a file.

2.1.1 Current User Stream

A required stream whose name MUST be "Current User".

The contents of this stream are specified by the **CurrentUserAtom** record (section 2.3.2).

2.1.2 PowerPoint Document Stream

A required stream whose name MUST be "PowerPoint Document".

Let a *top-level record* be specified as any one of the following: **DocumentContainer** (section 2.4.1), **MasterOrSlideContainer** (section 2.5.5), **HandoutContainer** (section 2.5.8), **SlideContainer** (section 2.5.1), **NotesContainer** (section 2.5.6), **ExOleObjStg** (section 2.10.34), **ExControlStg** (section 2.10.37), **VbaProjectStg** (section 2.10.40), **PersistDirectoryAtom** (section 2.3.4), or **UserEditAtom** (section 2.3.3) record.

The contents of this stream are specified by a sequence of *top-level records*. Partial ordering restrictions on the record sequence are specified in the **PersistDirectoryAtom** and **UserEditAtom** records.

As **container records**, the **DocumentContainer**, **MainMasterContainer** (section 2.5.3), **HandoutContainer** (section 2.5.8), **SlideContainer** (section 2.5.1), and **NotesContainer** (section 2.5.6) records are each the root of a tree of container records and **atom records**. Inside any container record, other records MAY exist that are not explicitly listed as child records. Unknown records are identified when the **recType** field of the **RecordHeader** structure (section 2.3.1) contains a value not specified by the **RecordType** enumeration (section 2.13.24). These unknown records, if encountered, MUST be ignored, and MAY<1> be preserved. Unknown records can be ignored by seeking forward **recLen** bytes from the end of the **RecordHeader** structure.

Each time this stream is written, new *top-level records*, a **user edit**, can be appended to the existing stream, or the entire stream contents can be replaced with an updated sequence of *top-level records*. If the entire stream is not replaced, any previously existing *top-level records* that comprised any previous user edit, can be made obsolete by the subsequently appended *top-level records* that comprise the current user edit.

Let a *live record* be specified as any *top-level record* in this stream, or any descendant of a *top-level record* in this stream, identified by the following process:

Part 1: Construct the **persist object directory**.

1. Read the **CurrentUserAtom** record (section 2.3.2) from the **Current User Stream** (section 2.1.1). All seek operations in the steps that follow this step are in the **PowerPoint Document Stream**.

2. Seek, in the **PowerPoint Document Stream**, to the offset specified by the **offsetToCurrentEdit** field of the **CurrentUserAtom** record identified in step 1.

3. Read the **UserEditAtom** record at the current offset. Let this record be a *live record*.

4. Seek to the offset specified by the **offsetPersistDirectory** field of the **UserEditAtom** record identified in step 3.

5. Read the **PersistDirectoryAtom** record at the current offset. Let this record be a *live record*.

6. Seek to the offset specified by the **offsetLastEdit** field in the **UserEditAtom** record identified in step 3.

7. Repeat steps 3 through 6 until **offsetLastEdit** is 0x00000000.

8. Construct the complete persist object directory for this file as follows:

    1. For each **PersistDirectoryAtom** record previously identified in step 5, add the **persist object identifier** and **persist object** stream offset pairs to the persist object directory starting with the **PersistDirectoryAtom** record last identified, that is, the one closest to the beginning of the stream.

    2. Continue adding these pairs to the persist object directory for each **PersistDirectoryAtom** record in the reverse order that they were identified in step 5; that is, the pairs from the **PersistDirectoryAtom** record closest to the end of the stream are added last.

    3. When adding a new pair to the persist object directory, if the persist object identifier already exists in the persist object directory, the persist object stream offset from the new pair replaces the existing persist object stream offset for that persist object identifier.

Part 2: Identify the document persist object.

1. Read the **docPersistIdRef** field of the **UserEditAtom** record first identified in step 3 of Part 1, that is, the **UserEditAtom** record closest to the end of the stream.

2. Lookup the value of the **docPersistIdRef** field in the persist object directory constructed in step 8 of Part 1 to find the stream offset of a persist object.

3. Seek to the stream offset specified in step 2.

4. Read the **DocumentContainer** record at the current offset. Let this record be a *live record*.

Part 3: Identify the notes master slide persist object.

1. Read the **documentAtom.notesMasterPersistIdRef** field of the **DocumentContainer** record identified in step 4 of Part 2. If the value of the field is zero, skip to step 1 of Part 4.

2. Lookup the value of the **documentAtom.notesMasterPersistIdRef** field in the persist object directory constructed in step 8 of Part 1 to find the stream offset of a persist object.

3. Seek to the stream offset specified in step 2.

4. Read the **NotesContainer** record at the current offset. Let this record be a *live record*.

Part 4: Identify the handout master slide persist object.

1. Read the **documentAtom.handoutMasterPersistIdRef** field of the **DocumentContainer** record identified in step 4 of Part 2. If the value of the field is zero, skip to step 1 of Part 5.

2. Lookup the value of the **documentAtom.handoutMasterPersistIdRef** field in the persist object directory constructed in step 8 of Part 1 to find the stream offset of a persist object.

3. Seek to the stream offset specified in step 2.

4. Read the **HandoutContainer** record at the current offset. Let this record be a *live record*.

Part 5: Identify the main master slide and title master slide persist objects.

1. Read the **MasterListWithTextContainer** record specified by the **masterList** field of the **DocumentContainer** record identified in step 4 of Part 2.

2. Read the first **MasterPersistAtom** (section 2.4.14.2) child record of the **MasterListWithTextContainer** record identified in step 1.

3. Lookup the value of the **persistIdRef** field of the **MasterPersistAtom** record previously identified in the persist object directory constructed in step 8 of Part 1 to find the stream offset of a persist object.

4. Seek to the stream offset specified in step 3.

5. Read the **MasterOrSlideContainer** record at the current offset. Let this record be a *live record*.

6. Repeat steps 3 through 5 for each **MasterPersistAtom** child record of the **MasterListWithTextContainer** record identified in step 1.

Part 6: Identify the presentation slide persist objects.

1. Read the **SlideListWithTextContainer** record (section 2.4.14.3), if present, specified by the **slideList** field of the **DocumentContainer** record identified in step 4 of Part 2. If not present, skip to step 1 of Part 7.

2. Read the first **SlidePersistAtom** (section 2.4.14.5) child record of the **SlideListWithTextContainer** record identified in step 1.

3. Lookup the value of the **persistIdRef** field of the **SlidePersistAtom** record (section 2.4.14.5) previously identified in the persist object directory constructed in step 8 of Part 1 to find the stream offset of a persist object.

4. Seek to the stream offset specified in step 3.

5. Read the **SlideContainer** record at the current offset. Let this record be a *live record*.

6. Repeat steps 3 through 5 for each **SlidePersistAtom** child record (section 2.4.14.5) of the **SlideListWithTextContainer** record identified in step 1.

Part 7: Identify the notes slide persist objects.

1. Read the **NotesListWithTextContainer** record (section 2.4.14.6), if present, specified by the **notesList** field of the **DocumentContainer** record identified in step 4 of Part 2. If not present, skip to step 1 of Part 8.

2. Read the first **NotesPersistAtom** (section 2.4.14.7) child record of the **NotesListWithTextContainer** record identified in step 1.

3. Lookup the value of the **persistIdRef** field of the **NotesPersistAtom** record previously identified in the persist object directory constructed in step 8 of Part 1 to find the stream offset of a persist object.

4. Seek to the stream offset specified in step 3.

5. Read the **NotesContainer** record at the current offset. Let this record be a *live record*.

6. Repeat steps 3 through 5 for each **NotesPersistAtom** child record of the **NotesListWithTextContainer** record identified in step 1.

Part 8: Identify the **ActiveX control** persist objects.

1. Read the **ExObjListContainer** record (section 2.10.1), if present, specified by the **exObjList** field of the **DocumentContainer** record identified in step 4 of Part 2. If not present, skip to step 1 of Part 11.

2. Read the first, if any, **ExControlContainer** child record (section 2.10.10) of the **ExObjListContainer** record identified in step 1. If no such child record exists, skip to step 1 of Part 9.

3. Lookup the value of the **exOleObjAtom.persistIdRef** field of the **ExControlContainer** record previously identified in the persist object directory constructed in step 8 of Part 1 to find the stream offset of a persist object.

4. Seek to the stream offset specified in step 3.

5. Read the **ExControlStg** record at the current offset. Let this record be a *live record*.

6. Repeat steps 3 through 5 for each **ExControlContainer** child record of the **ExObjListContainer** record identified in step 1.

Part 9: Identify the embedded OLE object persist objects.

1. Read the first, if any, **ExOleEmbedContainer** child record (section 2.10.27) of the **ExObjListContainer** record identified in step 1 of Part 8. If no such child record exists, skip to step 1 of Part 10.

2. Lookup the value of the **exOleObjAtom.persistIdRef** field of the **ExOleEmbedContainer** record previously identified in the persist object directory constructed in step 8 of Part 1 to find the stream offset of a persist object.

3. Seek to the stream offset specified in step 2.

4. Read the **ExOleObjStg** record at the current offset. Let this record be a *live record*.

5. Repeat steps 2 through 4 for each **ExOleEmbedContainer** child record of the **ExObjListContainer** record identified in step 1 of Part 8.

Part 10: Identify the linked OLE object persist objects.

1. Read the first, if any, **ExOleLinkContainer** child record (section 2.10.29) of the **ExObjListContainer** record identified in step 1 of Part 8. If no such child record exists, skip to step 1 of Part 11.

2. Lookup the value of the **exOleObjAtom.persistIdRef** field of the **ExOleLinkContainer** record previously identified in the persist object directory constructed in step 8 of Part 1 to find the stream offset of a persist object.

3. Seek to the stream offset specified in step 2.

4. Read the **ExOleObjStg** record at the current offset. Let this record be a *live record*.

5. Repeat steps 2 through 4 for each **ExOleLinkContainer** child record of the **ExObjListContainer** record identified in step 1 of Part 8.

Part 11: Identify the VBA project persist object.

1. Read the **DocInfoListContainer** record (section 2.4.4), if present, specified by the **docInfoList** field of the **DocumentContainer** record identified in step 4 of Part 2. If not present, skip to step 6.

2. Read the **VBAInfoContainer** (section 2.4.10) child record, if present, of the **DocInfoListContainer** record identified in step 1. If no such child record exists, skip to step 6.

3. Lookup the value of the **vbaInfoAtom.persistIdRef** field of the **VBAInfoContainer** record identified in step 2 in the persist object directory constructed in step 8 of Part 1 to find the stream offset of a persist object.

4. Seek to the stream offset specified in step 3.

5. Read the **VbaProjectStg** record at the current offset. Let this record be a *live record*.

6. End of process. All *live records* have been identified.

Let a *dead record* be specified as any *top-level record* in this stream, or any descendant of a *top-level record* in this stream, that is not a *live record*.

All uses of prescriptive terminology (MAY, SHOULD, MUST, SHOULD NOT, MUST NOT) in the specification of records in the following sections apply only to *live records*. The contents of all *dead records* are undefined and MUST be ignored.

2.1.3 Pictures Stream

An optional stream whose name MUST be "Pictures".

The contents of this stream are specified by the **OfficeArtBStoreDelay** record as specified in [MS-ODRAW] section 2.2.21.

2.1.4 Summary Information Stream

An optional stream whose name MUST be "\005SummaryInformation", where \005 is the character with value 0x0005, not the string literal "\005". This stream SHOULD be omitted for **encrypted documents**.

The contents of this stream are specified in [MS-OSHARED] section 2.3.3.2.1.

2.1.5 Document Summary Information Stream

An optional stream whose name MUST be "\005DocumentSummaryInformation", where \005 is the character with value 0x0005, not the string literal "\005". This stream MAY<2> be omitted for encrypted documents.

The contents of this stream are specified in [MS-OSHARED] section 2.3.3.2.2.

2.1.6 Encrypted Summary Information Stream

An optional stream whose name MUST be "EncryptedSummary". This stream exists only in an encrypted document.

The contents of this stream are specified in [MS-OFFCRYPTO] section 2.3.5.4.

2.1.7 Digital Signature Storage

An optional storage whose name MUST be "_xmlsignatures". It MAY<3> be omitted and MAY be ignored.

The contents of this storage are specified in [MS-OFFCRYPTO] section 2.5.2.

2.1.8 Custom XML Data Storage

An optional storage whose name MUST be "MsoDataStore".

The contents of the storage are specified in [MS-OSHARED] section 2.3.6.

2.1.9 Signatures Stream

An optional stream whose name MUST be "_signatures". It SHOULD<4> be omitted and MAY be ignored.

The contents of this stream are specified in [MS-OFFCRYPTO] section 2.5.1.

2.2 Basic Types

2.2.1 BlipRef

Referenced by: TextPFException9

A 2-byte signed integer that specifies a zero-based index of a picture bullet within the collection of picture bullets specified by the **BlipCollection9Container** record (section 2.9.72). The value 0xFFFF specifies a **null** reference.

2.2.2 bool1

Referenced by: BuildAtom, ChartBuildAtom, DiffRecordHeaders, DocumentAtom, ExOleEmbedAtom, NormalViewSetInfoAtom, NoZoomViewInfoAtom, ParaBuildAtom, PhotoAlbumInfo10Atom, PrintOptionsAtom, SlideViewInfoAtom, TimeScaleBehaviorAtom, TimeVariantBool, ZoomViewInfoAtom

A 1-byte unsigned integer that specifies a Boolean value. It SHOULD be 0x00 or 0x01. A value of 0x00 specifies **FALSE** and all other values specify **TRUE**.

2.2.3 BulletSize

Referenced by: TextPFException

A 2-byte signed integer that specifies the bullet font size. It MUST<5> be a value from the following table:

Range	Meaning
25 to 400, inclusive.	The value specifies bullet font size as a percentage of the font size of the first text run in the paragraph.
-4000 to -1, inclusive.	The absolute value specifies the bullet font size in **points**.

2.2.4 char2

Referenced by: BookmarkEntityAtom, FontEntityAtom, RTFDateTimeMCAtom, SlideShowDocInfoAtom

An array of bytes that specifies a UTF-16 Unicode [RFC2781] string. The Unicode **NULL** character (0x0000), if present, terminates the string.

2.2.5 ExHyperlinkId

Referenced by: ExHyperlinkAtom

A 4-byte unsigned integer that specifies an identifier for a **hyperlink**. It MUST be greater than 0x00000000. The combined set of **ExObjId** (section 2.2.7) and **ExHyperlinkId** values in the file MUST NOT contain duplicates.

2.2.6 ExHyperlinkIdRef

Referenced by: ExHyperlinkRefAtom, InteractiveInfoAtom

A 4-byte unsigned integer that specifies a reference to a hyperlink. It MUST be 0x00000000 or equal to the value of the **exHyperlinkId** field of an **ExHyperlinkAtom** record (section 2.10.17). The value 0x00000000 specifies a **null** reference.

2.2.7 ExObjId

Referenced by: ExMediaAtom, ExOleObjAtom

A 4-byte unsigned integer that specifies an identifier for an external object. It MUST be greater than 0x00000000. The combined set of **ExObjId** and **ExHyperlinkId** (section 2.2.5) values in the file MUST NOT contain duplicates.

2.2.8 ExObjIdRef

Referenced by: ExObjRefAtom

A 4-byte unsigned integer that specifies a reference to an external object. It MUST be equal to the value of the **exObjId** field of an **ExMediaAtom** record (section 2.10.6) or the value of the **exObjId** field of an **ExOleObjAtom** record (section 2.10.12).

2.2.9 FileOrDirNameFragment

Referenced by: BCBroadcastDateTimeAtom, BCPresentationNameAtom, BCUserNameAtom

An array of bytes that specifies a UTF-16 Unicode [RFC2781] string that specifies a valid file name or directory name. See [MSDN-FILE] for more information about file naming.

2.2.10 FontIndexRef

Referenced by: TextCFException, TextPFException

A 2-byte unsigned integer that specifies a zero-based index of a font within the collection of fonts specified by the **FontCollectionContainer** record (section 2.9.8).

2.2.11 FontIndexRef10

Referenced by: TextCFException10

A 2-byte unsigned integer that specifies a zero-based index of a font within the collection of fonts specified by the **FontCollection10Container** record (section 2.9.11).

2.2.12 HttpUrl

Referenced by: BCChatUrlAtom, SlideLibUrlAtom

An array of bytes that specifies a UTF-16 Unicode [RFC2781] string. It MUST be a valid URI [RFC3986] with the HTTP scheme.

2.2.13 IndentLevel

Referenced by: MasterTextPropRun, TextPFRun

A 2-byte unsigned integer that specifies a text paragraph indent level. It MUST be less than or equal to 0x0004.

2.2.14 MachineName

Referenced by: BCNetShowServerNameAtom, BCRexServerNameAtom

An array of bytes that specifies a UTF-16 Unicode [RFC2781] string that specifies a **computer name**. See [MSDN-FILE] for more information about computer naming.

2.2.15 MarginOrIndent

Referenced by: TextPFException, TextRuler

A 2-byte signed integer that specifies an offset in **master units**. The origin of the offset is specified at each use of this **MarginOrIndent** type.

2.2.16 MasterId

Referenced by: MasterPersistAtom

A 4-byte unsigned integer that specifies an identifier for a main master slide or title master slide. It MUST be greater than or equal to 0x80000000. The set of **MasterId** values in the file MUST NOT contain duplicates.

2.2.17 MasterIdRef

Referenced by: SlideAtom

A 4-byte unsigned integer that specifies a reference to a main master slide or title master slide. It MUST be 0x00000000 or equal to the value of the **masterId** field of a **MasterPersistAtom** record (section 2.4.14.2). The value 0x00000000 specifies a **null** reference.

2.2.18 NotesId

Referenced by: NotesPersistAtom

A 4-byte unsigned integer that specifies an identifier for a notes slide. It MUST be greater than or equal to 0x00000100 and MUST be less than or equal to 0x7FFFFFFF. The set of **NotesId** values in the file MUST NOT contain duplicates.

2.2.19 NotesIdRef

Referenced by: SlideAtom

A 4-byte unsigned integer that specifies a reference to a notes slide. It MUST be 0x00000000 or equal to the value of the **notesId** field of a **NotesPersistAtom** record (section 2.4.14.7). The value 0x00000000 specifies a **null** reference.

2.2.20 ParaSpacing

Referenced by: TextPFException

A 2-byte signed integer that specifies text paragraph spacing. It MUST be a value from the following table:

Range	Meaning
0 to 13200, inclusive.	The value specifies spacing as a percentage of the text line height.
Less than 0.	The absolute value specifies spacing in master units.

2.2.21 PersistIdRef

Referenced by: DocumentAtom, ExOleObjAtom, MasterPersistAtom, NotesPersistAtom, SlidePersistAtom, UserEditAtom, VBAInfoAtom

A 4-byte unsigned integer that specifies a reference to a persist object. It MUST be 0x00000000 or equal to a persist object identifier specified by a **PersistDirectoryAtom** record (section 2.3.4). The value 0x00000000 specifies a **null** reference.

2.2.22 PrintableAnsiString

Referenced by: CurrentUserAtom, DocRoutingSlipString

An array of bytes that specifies an ANSI string. It MUST NOT contain the following characters:

0x00 - 0x1F

0x7F - 0x9F

The ANSI **NULL** character (0x00), if present, terminates the string.

2.2.23 PrintableUnicodeString

Referenced by: AuthorNameAtom, BCContactAtom, BCEmailAddressAtom, BCEmailNameAtom, BCSpeakerAtom, BookmarkValueAtom, ClipboardNameAtom, Comment10AuthorAtom, Comment10AuthorInitialAtom, CopyrightAtom, CurrentUserAtom, FileNameAtom, FooterAtom, HeaderAtom, KeywordsAtom, MacroNameAtom, ModifyPasswordAtom, NamedShowAtom, NamedShowNameAtom, PP10DocBinaryTagExtension, PP10ShapeBinaryTagExtension, PP10SlideBinaryTagExtension, PP11DocBinaryTagExtension, PP11ShapeBinaryTagExtension, PP12DocBinaryTagExtension, PP12SlideBinaryTagExtension, PP9DocBinaryTagExtension, PP9ShapeBinaryTagExtension, PP9SlideBinaryTagExtension, ProgIDAtom, ReviewerNameAtom, ServerIdAtom, SoundNameAtom, TagNameAtom, UserDateAtom

An array of bytes that specifies a UTF-16 Unicode [RFC2781] string. It MUST NOT contain the following characters:

- 0x0000 - 0x001F
- 0x007F - 0x009F

The Unicode **NULL** character (0x0000), if present, terminates the string.

2.2.24 SlideId

Referenced by: SlidePersistAtom

A 4-byte unsigned integer that specifies an identifier for a presentation slide. It MUST be greater than or equal to 0x00000100 and MUST be less than or equal to 0x7FFFFFFF. The set of **SlideId** values in the file MUST NOT contain duplicates.

2.2.25 SlideIdRef

Referenced by: ExControlAtom, ExOleLinkAtom, LinkedSlide10Atom, NamedShowSlidesAtom, NotesAtom, OutlineTextPropsHeaderExAtom, SlideListEntry10Atom, UserEditAtom

A 4-byte unsigned integer that specifies a reference to a presentation slide. It MUST be 0x00000000 or equal to the value of the **slideId** field of a **SlidePersistAtom** record (section 2.4.14.5). The value 0x00000000 specifies a **null** reference.

2.2.26 SmartTagIndex

Referenced by: SmartTags

A 4-byte unsigned integer that specifies a zero-based index of a **smart tag** within the collection of smart tags specified by the **SmartTagStore11Container** record (section 2.11.28).

2.2.27 SoundIdRef

Referenced by: AnimationInfoAtom, ExWAVAudioEmbeddedAtom, InteractiveInfoAtom, SlideShowSlideInfoAtom, VisualSoundAtom

A 4-byte unsigned integer that specifies a reference to a sound. It MUST be 0x00000000 or equal to the integer value of the **soundIdAtom** field of a **SoundContainer** record (section 2.4.16.3). The value 0x00000000 specifies a **null** reference.

2.2.28 TabCrLfPrintableUnicodeString

Referenced by: Comment10TextAtom

An array of bytes that specifies a UTF-16 Unicode [RFC2781] string. It MUST NOT contain the following characters:

- 0x0000 - 0x0008
- 0x000B
- 0x000C
- 0x000E - 0x001F
- 0x007F - 0x009F

The Unicode **NULL** character (0x0000), if present, terminates the string.

2.2.29 TabSize

Referenced by: TextPFException, TextRuler

A 2-byte signed integer that specifies the size, in master units, of a tab.

2.2.30 TextPosition

Referenced by: DateTimeMCAtom, FooterMCAtom, GenericDateMCAtom, HeaderMCAtom, RTFDateTimeMCAtom, SlideNumberMCAtom, TextBookmarkAtom, TextRange

A 4-byte signed integer that specifies a zero-based character position in a range of text. It MUST be greater than or equal to 0x00000000, and MUST be less than the character length of the *corresponding text*.

Let the *corresponding text* be as specified at each use of this **TextPosition** type.

2.2.31 TxLCID

Referenced by: TextSIException

A 2-byte unsigned integer that specifies a language identifier. It MUST be a value from the following table.

Value	Meaning
0x0000	No language.
0x0013	Any Dutch language is preferred over non-Dutch languages when proofing the text.
0x0400	No proofing is performed on the text.
Greater than 0x0400	A valid LCID as specified by [MS-LCID].

2.2.32 UncOrLocalPath

Referenced by: BCArchiveDirAtom, UncOrLocalPathAtom

An array of bytes that specifies a UTF-16 Unicode [RFC2781] string that specifies a **UNC** or local file system path. See [MSDN-FILE] for more information about file naming.

2.2.33 UncPath

Referenced by: BCAsdFileNameAtom, BCNetShowFilesBaseDirAtom, BCNetShowFilesDirAtom, BCPptFilesBaseDirAtom, BCPptFilesDirAtom

An array of bytes that specifies a UTF-16 Unicode [RFC2781] string that specifies a UNC path. See [MSDN-FILE] for more information about file naming.

2.2.34 UncPathOrHttpUrl

Referenced by: BCPptFilesBaseUrlAtom

An array of bytes that specifies a UTF-16 Unicode [RFC2781] string that specifies a UNC path or a valid URI [RFC3986] with the HTTP scheme. See [MSDN-FILE] for more information about file naming.

2.2.35 UnicodeString

Referenced by: BCDescriptionAtom, BCEntryIDAtom, BCTitleAtom, FriendlyNameAtom, LocationAtom, MenuNameAtom, ScreenTipAtom, SlideNameAtom, TagValueAtom, TargetAtom, TemplateNameAtom, TimeEventFilter, TimeNodeTimeFilter, TimePointsTypes, TimeRuntimeContext, TimeVariantString

An array of bytes that specifies a UTF-16 Unicode [RFC2781] string. The Unicode **NULL** character (0x0000), if present, terminates the string.

2.2.36 Utf8UnicodeString

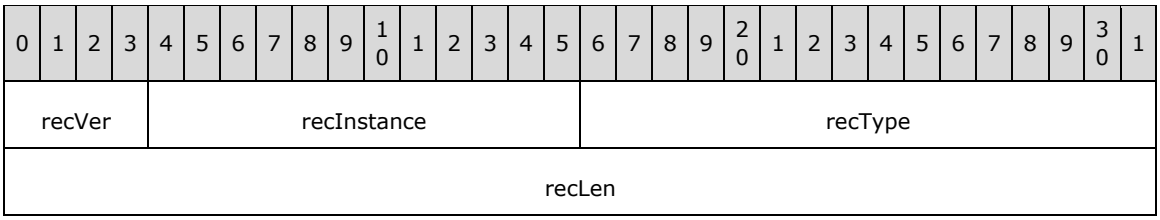
Referenced by: RoundTripColorMappingAtom

An array of bytes that specifies a UTF-8 Unicode [RFC3629] string. It MUST be valid XML as defined in [XML].

2.3 File Structure Types

2.3.1 RecordHeader

A structure at the beginning of each container record and each atom record in the file. The values in the record header and the context of the record are used to identify and interpret the record data that follows.



recVer (4 bits): An unsigned integer that specifies the version of the record data that follows the record header. A value of 0xF specifies that the record is a container record.

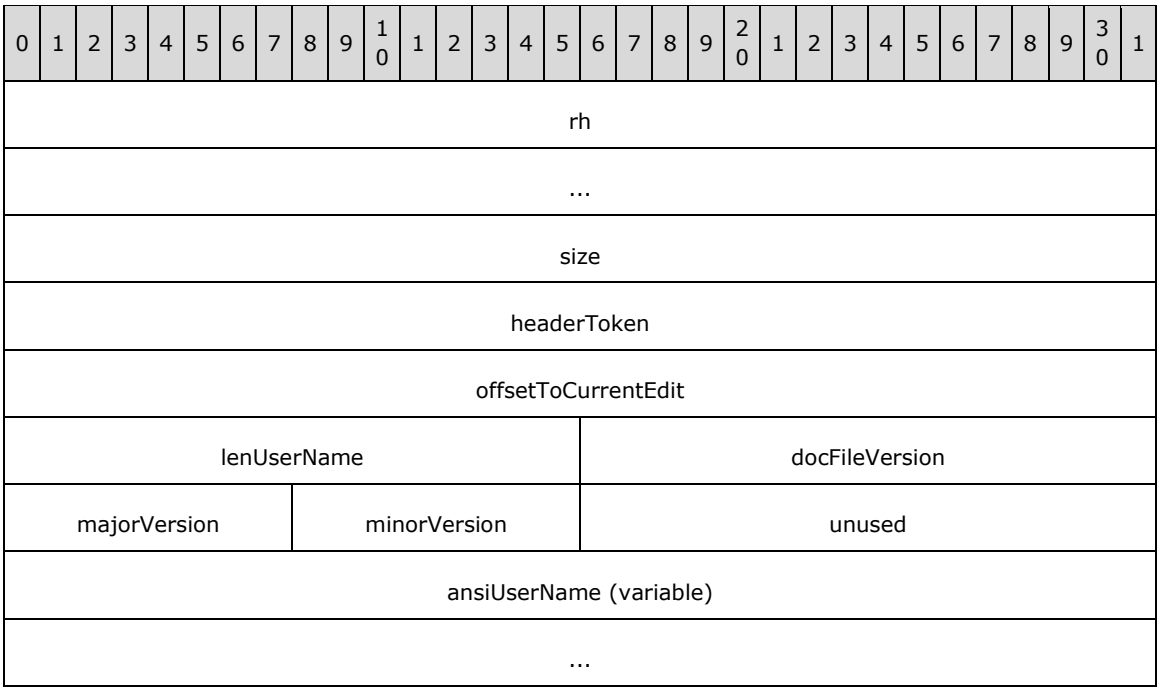
recInstance (12 bits): An unsigned integer that specifies the record instance data. Interpretation of the value is dependent on the particular record type.

recType (2 bytes): A **RecordType** enumeration (section 2.13.24) that specifies the type of the record data that follows the record header.

recLen (4 bytes): An unsigned integer that specifies the length, in bytes, of the record data that follows the record header.

2.3.2 CurrentUserAtom

An atom record that specifies information about the last user to modify the file and where the most recent user edit is located. This is the only record in the **Current User Stream** (section 2.1.1).



relVersion
unicodeUserName (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_CurrentUserAtom** (section 2.13.24).

size (4 bytes): An unsigned integer that specifies the length, in bytes, of the fixed-length portion of the record, which begins after the **rh** field and ends before the **ansiUserName** field. It MUST be 0x00000014.

headerToken (4 bytes): An unsigned integer that specifies a token used to identify whether the file is encrypted. It MUST be a value from the following table.

Value	Meaning
0xE391C05F	The file SHOULD NOT<6> be an encrypted document.
0xF3D1C4DF	The file MUST be an encrypted document.

offsetToCurrentEdit (4 bytes): An unsigned integer that specifies an offset, in bytes, from the beginning of the **PowerPoint Document Stream** (section 2.1.2) to the **UserEditAtom** record (section 2.3.3) for the most recent user edit.

lenUserName (2 bytes): An unsigned integer that specifies the length, in bytes, of the **ansiUserName** field. It MUST be less than or equal to 255.

docFileVersion (2 bytes): An unsigned integer that specifies the document file version of the file. It MUST be 0x03F4.

majorVersion (1 byte): An unsigned integer that specifies the major version of the storage format. It MUST be 0x03.

minorVersion (1 byte): An unsigned integer that specifies the minor version of the storage format. It MUST be 0x00.

unused (2 bytes): Undefined and MUST be ignored.

ansiUserName (variable): A **PrintableAnsiString** (section 2.2.22) that specifies the user name of the last user to modify the file. The length, in bytes, of the field is specified by the **lenUserName** field.

relVersion (4 bytes): An unsigned integer that specifies the release version of the file format. It MUST be a value from the following table.

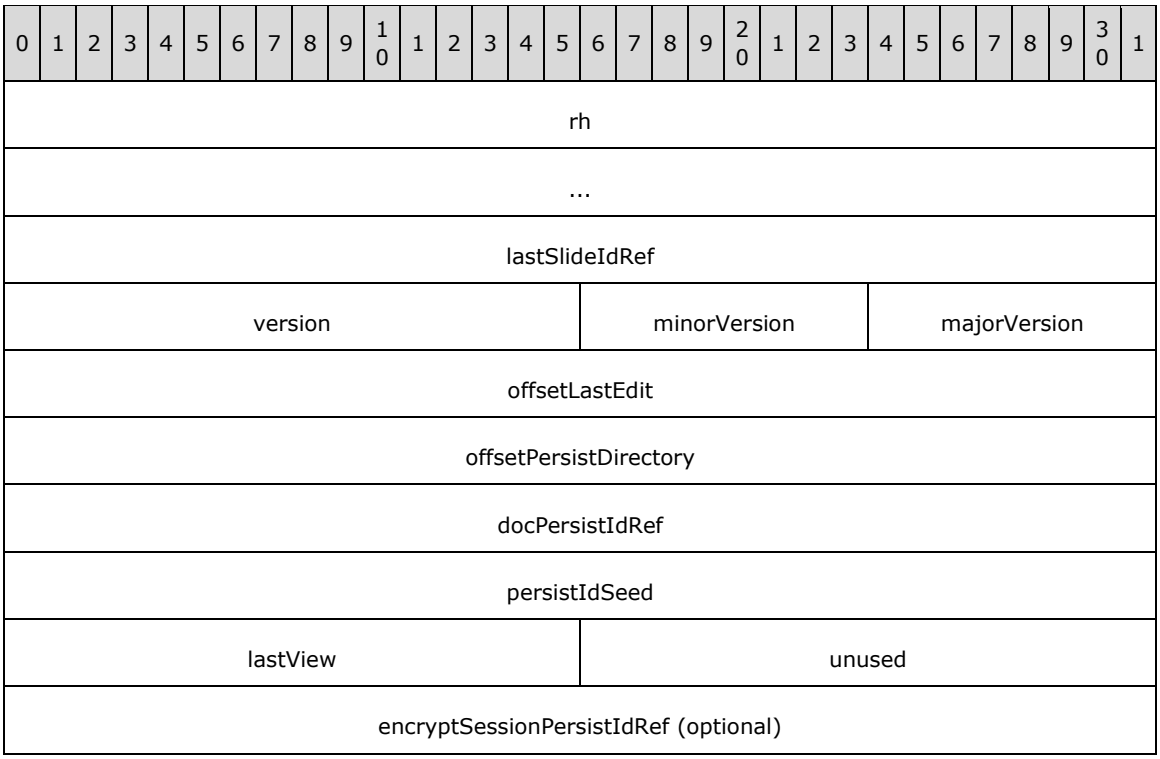
Value	Meaning
0x00000008	The file contains one or more main master slide.

Value	Meaning
0x00000009	The file contains more than one main master slide. It SHOULD NOT<7> be used.

unicodeUserName (variable): An optional **PrintableUnicodeString** (section 2.2.23) that specifies the user name of the last user to modify the file. The length, in bytes, of the field is specified by 2 * **lenUserName**. This user name supersedes that specified by the **ansiUserName** field. It MAY<8> be omitted.

2.3.3 UserEditAtom

An atom record that specifies information about a user edit.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_UserEditAtom** (section 2.13.24).
rh.recLen	MUST be 0x0000001C or 0x00000020.

lastSlideIdRef (4 bytes): A **SlideIdRef** (section 2.2.25) that specifies the last slide viewed, if this is the last **UserEditAtom** record in the **PowerPoint Document Stream** (section 2.1.2). In all other cases the value of this field is undefined and MUST be ignored.

version (16 bits): An unsigned integer that specifies a build version of the executable that wrote the file. It SHOULD<9> be 0x0000 and MUST be ignored.

minorVersion (8 bits): An unsigned integer that specifies the minor version of the storage format. It MUST be 0x00.

majorVersion (8 bits): An unsigned integer that specifies the major version of the storage format. It MUST be 0x03.

offsetLastEdit (4 bytes): An unsigned integer that specifies an offset, in bytes, from the beginning of the **PowerPoint Document Stream** to a **UserEditAtom** record for the previous user edit. It MUST be less than the offset, in bytes, of this **UserEditAtom** record. The value 0x00000000 specifies that no previous user edit exists.

offsetPersistDirectory (4 bytes): An unsigned integer that specifies an offset, in bytes, from the beginning of the **PowerPoint Document Stream** to the **PersistDirectoryAtom** record (section 2.3.4) for this user edit. It MUST be greater than **offsetLastEdit** and less than the offset, in bytes, of this **UserEditAtom** record.

docPersistIdRef (4 bytes): A **PersistIdRef** (section 2.2.21) that specifies the value to look up in the persist object directory to find the offset of the **DocumentContainer** record (section 2.4.1). It MUST be 0x00000001.

persistIdSeed (4 bytes): An unsigned integer that specifies a **seed** for creating a new persist object identifier. It MUST be greater than or equal to all persist object identifiers in the file as specified by the **PersistDirectoryAtom** records.

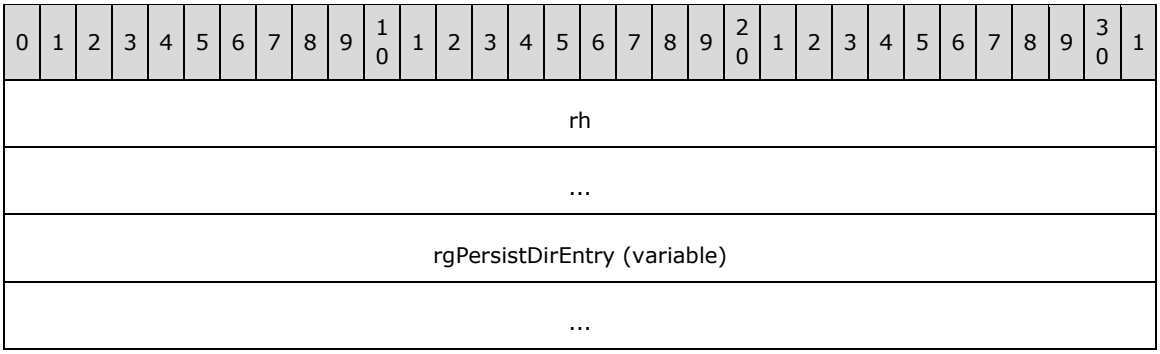
lastView (2 bytes): A **ViewTypeEnum** enumeration (section 2.13.42) that specifies the last view used to display the file.

unused (2 bytes): Undefined and MUST be ignored.

encryptSessionPersistIdRef (4 bytes): An optional **PersistIdRef** that specifies the value to look up in the persist object directory to find the offset of the **CryptSession10Container** record (section 2.3.7). It MAY<10> be omitted. It MUST exist if the document is an encrypted document.

2.3.4 PersistDirectoryAtom

An atom record that specifies a persist object directory. Each persist object identifier specified MUST be unique in that persist object directory.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.

Field	Meaning
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_PersistDirectoryAtom** (section 2.13.24).

rgPersistDirEntry (variable): An array of **PersistDirectoryEntry** structures (section 2.3.5) that specifies persist object identifiers and stream offsets to persist objects. The size, in bytes, of the array is specified by **rh.recLen**.

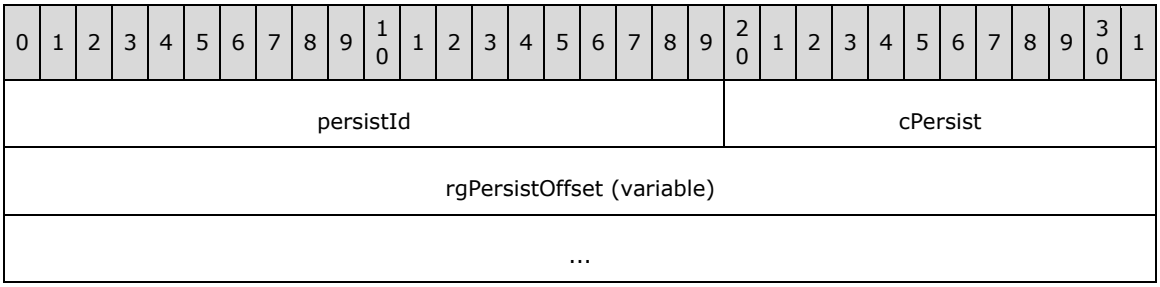
2.3.5 PersistDirectoryEntry

Referenced by: PersistDirectoryAtom

A structure that specifies a compressed table of sequential persist object identifiers and stream offsets to associated persist objects.

Let the *corresponding user edit* be specified by the **UserEditAtom** record (section 2.3.3) that most closely follows the **PersistDirectoryAtom** record (section 2.3.4) that contains this structure.

Let the *corresponding persist object directory* be specified by the **PersistDirectoryAtom** record that contains this structure.



persistId (20 bits): An unsigned integer that specifies a starting persist object identifier. It MUST be less than or equal to 0xFFFFE. The first entry in **rgPersistOffset** is associated with **persistId.** The next entry, if present, is associated with **persistId** plus 1. Each entry in **rgPersistOffset** is associated with a persist object identifier in this manner, with the final entry associated with `persistId + cPersist – 1`.

cPersist (12 bits): An unsigned integer that specifies the count of items in **rgPersistOffset**. It MUST be greater than or equal to 0x001.

rgPersistOffset (variable): An array of **PersistOffsetEntry** (section 2.3.6) that specifies stream offsets to persist objects. The count of items in the array is specified by **cPersist**. The value of each item MUST be greater than or equal to **offsetLastEdit** in the *corresponding user edit* and MUST be less than the offset, in bytes, of the *corresponding persist object directory*.

2.3.6 PersistOffsetEntry

Referenced by: PersistDirectoryEntry

An unsigned 4-byte integer that specifies an offset, in bytes, from the beginning of the **PowerPoint Document Stream** (section 2.1.2) to a persist object.

2.3.7 CryptSession10Container

A container record that specifies encryption properties for the file.

Only the Office Binary Document RC4 CryptoAPI encryption method as specified in [MS-OFFCRYPTO] section 2.3.5 is supported.

The following conditions MUST apply for the **Current User Stream** (section 2.1.1) of an encrypted file:

- The stream MUST NOT be encrypted.
- The **headerToken** field of the **CurrentUserAtom** record (section 2.3.2) SHOULD<11> be equal to 0xF3D1C4DF.

The following conditions MUST apply for the **PowerPoint Document Stream** (section 2.1.2) of an encrypted file:

- The **UserEditAtom** record (section 2.3.3) and the **PersistDirectoryAtom** record (section 2.3.4) MUST NOT be encrypted.
- The **rh** field of this **CryptSession10Container** record MUST NOT be encrypted.
- The **data** field of this **CryptSession10Container** record MUST be encrypted as specified in [MS-OFFCRYPTO] section 2.3.5.1.
- All other parts of the stream MUST be encrypted.
- The stream MUST contain exactly one **UserEditAtom** record.
- The **encryptSessionPersistIdRef** field of the **UserEditAtom** record MUST exist. It MUST refer to a valid persist object, which MUST contain this **CryptSession10Container** record.

The **Pictures Stream** (section 2.1.3) MUST be encrypted.

The following conditions MUST apply for the **Summary Info Stream** (section 2.1.4) and **Document Summary Info Stream** (section 2.1.5) of an encrypted file if the **fDocProps** bit is not set in the **data.EncryptionHeader.Flags** field:

- The **Summary Info Stream** MUST NOT exist.
- The **Encrypted Summary Info Stream** (section 2.1.6) MUST exist.
- The **Document Summary Info Stream** SHOULD exist but MUST be empty.

If the **fDocProps** bit is set in the **data.EncrytionHeader.Flags** field, the **Summary Info Stream** and **Document Summary Info Stream** MUST NOT be encrypted and MUST contain the same content as the unencrypted file.

Encrypted parts of an encrypted document MUST be decrypted according to the following rules:

- A password hash MUST be generated from a password and the **data.EncryptionVerifier.Salt** field as specified in [MS-OFFCRYPTO] section 2.3.5.5.
- The password MUST be the same as the password which was used to encrypt the file.
- For each block number the derived encryption key MUST be generated from the password hash and the block number as specified in [MS-OFFCRYPTO] section 2.3.5.2.
- The corresponding derived encryption key MUST be used to decrypt the data.

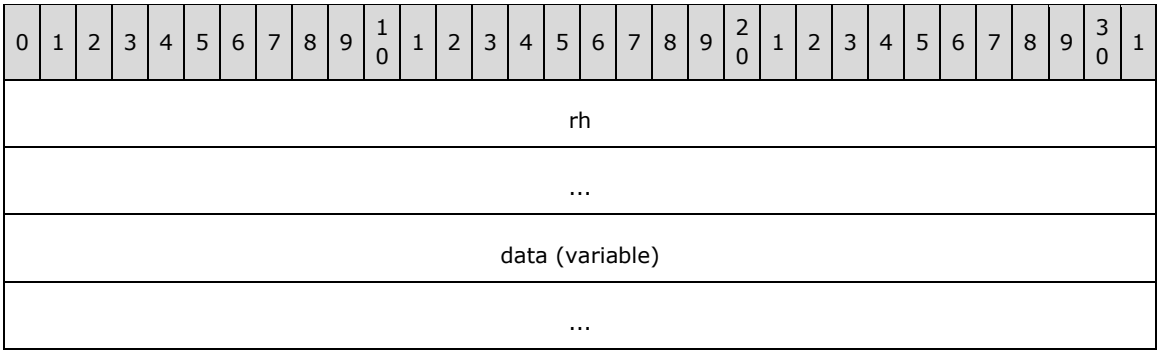
A persist object in the **PowerPoint Document Stream** is decrypted as follows:

- For a persist object, the block number for the derived encryption key is the persist object identifier.
- The derived encryption key for a persist object MUST be generated from the password hash and the persist object identifier.
- The range of bytes of the persist object that MUST be decrypted using the derived encryption key is specified by:
  - The file offset of the persist object as specified in the **PowerPoint Document Stream** section.
  - The length in bytes which is 8 + the **recLen** field of the **RecordHeader** structure (section 2.3.1) at the file offset.
- After decrypting, the range of bytes conforms to the content as specified by this document.

A picture in the **Pictures Stream** is decrypted as follows:

For each field of an **OfficeArtBStoreContainerFileBlock** record as specified in [MS-ODRAW] section 2.2.22 the following decryption steps apply:

- The derived encryption key MUST be generated from the password hash and a block number equal to zero.
- The size of each field in bytes MUST be decrypted with the derived encryption key.



rh (8 bytes): A **RecordHeader** structure that specifies the header for this record. Sub-fields are further specified in the following table:

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_CryptSession10Container** (section 2.13.24).

data (variable): An **EncryptionHeader** record as specified in [MS-OFFCRYPTO] section 2.3.5.1. The length, in bytes, of the field is specified by **rh.recLen.**

2.4 Document Types

2.4.1 DocumentContainer

A container record that specifies information about the document.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
documentAtom (48 bytes)																															
...																															
...																															
exObjList (variable)																															
...																															
documentTextInfo (variable)																															
...																															
soundCollection (variable)																															
...																															
drawingGroup (variable)																															
...																															
masterList (variable)																															
...																															
docInfoList (variable)																															
...																															
slideHF (variable)																															
...																															
notesHF (variable)																															
...																															
slideList (variable)																															
...																															
notesList (variable)																															

...	
slideShowDocInfoAtom (88 bytes, optional)	
...	
...	
namedShows (variable)	
...	
summary (variable)	
...	
docRoutingSlipAtom (variable)	
...	
printOptionsAtom (13 bytes, optional)	
...	
...	
...	rtCustomTableStylesAtom1 (variable)
...	
endDocumentAtom	
...	
rtCustomTableStylesAtom2 (variable)	
...	

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table:

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_Document** (section 2.13.24)

documentAtom (48 bytes): A **DocumentAtom** record (section 2.4.2) that specifies size information for presentation slides and notes slides.

exObjList (variable): An optional **ExObjListContainer** record (section 2.10.1) that specifies the list of external objects in the document.

documentTextInfo (variable): A **DocumentTextInfoContainer** record (section 2.9.1) that specifies the default text styles for the document.

soundCollection (variable): An optional **SoundCollectionContainer** record (section 2.4.16.1) that specifies the list of sounds in the file.

drawingGroup (variable): A **DrawingGroupContainer** record (section 2.4.3) that specifies drawing information for the document.

masterList (variable): A **MasterListWithTextContainer** record (section 2.4.14.1) that specifies the list of main master slides and title master slides.

docInfoList (variable): An optional **DocInfoListContainer** record (section 2.4.4) that specifies additional document information.

slideHF (variable): An optional **SlideHeadersFootersContainer** record (section 2.4.15.1) that specifies the default **header** and **footer** information for presentation slides.

notesHF (variable): An optional **NotesHeadersFootersContainer** record (section 2.4.15.6) that specifies the default header and footer information for notes slides.

slideList (variable): An optional **SlideListWithTextContainer** record (section 2.4.14.3) that specifies the list of presentation slides.

notesList (variable): An optional **NotesListWithTextContainer** record (section 2.4.14.6) that specifies the list of notes slides.

slideShowDocInfoAtom (88 bytes): An optional **SlideShowDocInfoAtom** record (section 2.6.1) that specifies slide show information for the document.

namedShows (variable): An optional **NamedShowsContainer** record (section 2.6.2) that specifies named shows in the document.

summary (variable): An optional **SummaryContainer** record (section 2.4.22.3) that specifies **bookmarks** for the document.

docRoutingSlipAtom (variable): An optional **DocRoutingSlipAtom** record (section 2.11.1) that specifies document routing information.

printOptionsAtom (13 bytes): An optional **PrintOptionsAtom** record (section 2.4.12) that specifies default print options.

rtCustomTableStylesAtom1 (variable): An optional **RoundTripCustomTableStyles12Atom** record (section 2.11.13) that specifies round-trip information for custom table styles.

endDocumentAtom (8 bytes): An **EndDocumentAtom** record (section 2.4.13) that specifies the end of the information for the document.

rtCustomTableStylesAtom2 (variable): An optional **RoundTripCustomTableStyles12Atom** record that specifies round-trip information for custom table styles. It MUST NOT exist if **rtCustomTableStylesAtom1** exists.

2.4.2 DocumentAtom

Referenced by: DocumentContainer

An atom record that specifies information about the entire document.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
slideSize																															
...																															
notesSize																															
...																															
serverZoom																															
...																															
notesMasterPersistIdRef																															
handoutMasterPersistIdRef																															
firstSlideNumber																	slideSizeType														
fSaveWithFonts											fOmitTitlePlace										fRightToLeft										fShowComments									

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x1.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_DocumentAtom** (section 2.13.24).
rh.recLen	MUST be 0x00000028.

slideSize (8 bytes): A **PointStruct** structure (section 2.12.5) that specifies the dimensions of the presentation slides in master units. Sub-fields are further specified in the following table.

Field	Meaning
slideSize.x	Specifies the width. It MUST be greater than or equal to 0x00000240 and less than or equal to 0x00007E00.
slideSize.y	Specifies the height. It MUST be greater than or equal to 0x00000240 and less than or equal to 0x00007E00.

notesSize (8 bytes): A **PointStruct** structure that specifies the dimensions of the notes slides and **handout slides** in master units. Sub-fields are further specified in the following table.

Field	Meaning
notesSize.x	Specifies the width. It MUST be greater than or equal to 0x00000240 and less than or equal to 0x00007E00.
notesSize.y	Specifies the height. It MUST be greater than or equal to 0x00000240 and less than or equal to 0x00007E00.

serverZoom (8 bytes): A **RatioStruct** structure (section 2.12.6) that specifies a zoom level for visual representations of the document in **Object Linking and Embedding (OLE)** scenarios. The ratio specified by this field MUST be greater than zero.

notesMasterPersistIdRef (4 bytes): A **PersistIdRef** (section 2.2.21) that specifies the value to look up in the persist object directory to find the offset of a **NotesContainer** record (section 2.5.6) that specifies the notes master slide.

handoutMasterPersistIdRef (4 bytes): A **PersistIdRef** that specifies the value to look up in the persist object directory to find the offset of a **HandoutContainer** record (section 2.5.8) that specifies the handout master slide.

firstSlideNumber (2 bytes): An **unsigned integer** that specifies the starting number for numbering slides. It MUST be less than 10000.

slideSizeType (2 bytes): A **SlideSizeEnum** enumeration (section 2.13.26) that specifies the type of a presentation slide size.

fSaveWithFonts (1 byte): A **bool1** (section 2.2.2) that specifies whether fonts are embedded in the document.

fOmitTitlePlace (1 byte): A **bool1** that specifies whether placeholder shapes on the title slide are not displayed.

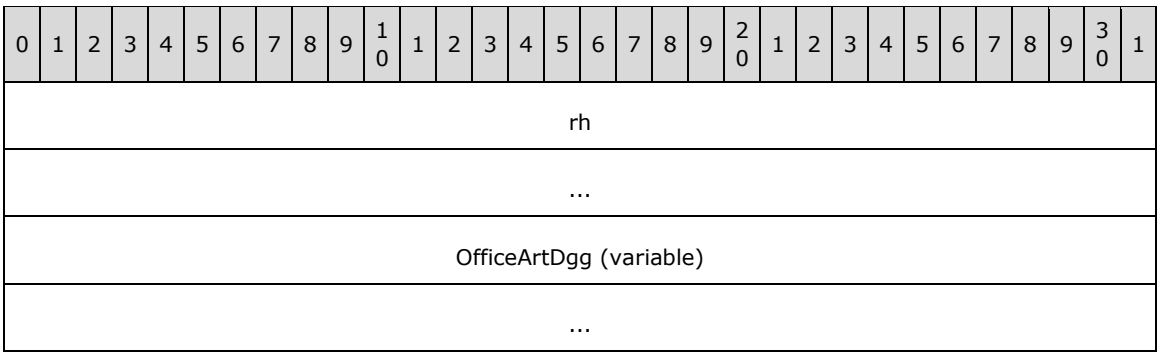
fRightToLeft (1 byte): A **bool1** that specifies whether the user interface displays the document optimized for right-to-left languages.

fShowComments (1 byte): A **bool1** that specifies whether **presentation comments** are displayed.

2.4.3 DrawingGroupContainer

Referenced by: DocumentContainer

A container record that specifies drawing information for the document.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table:

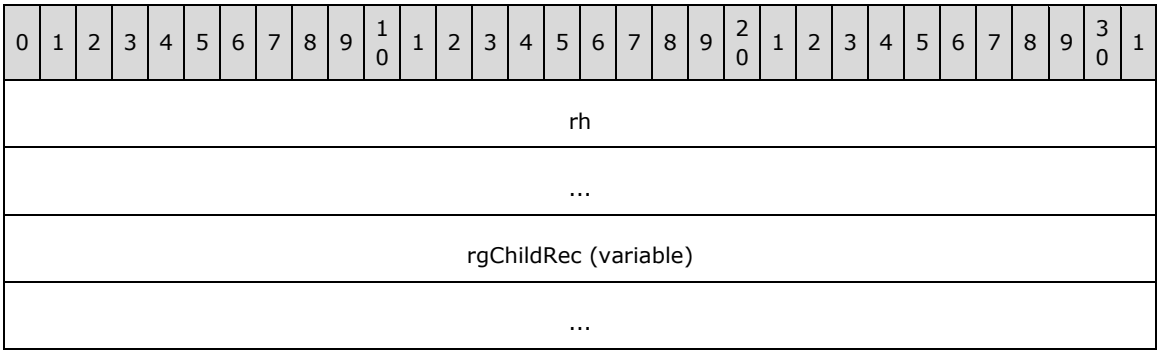
Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_DrawingGroup** (section 2.13.24).

OfficeArtDgg (variable): An **OfficeArtDggContainer** ([MS-ODRAW] section 2.2.12) that specifies drawing information for the document.

2.4.4 DocInfoListContainer

Referenced by: DocumentContainer

A container record that specifies information about the document and document display settings.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_List** (section 2.13.24).

rgChildRec (variable): An array of **DocInfoListSubContainerOrAtom** records (section 2.4.5) that specifies information about the document or how the document is displayed. The size, in bytes, of the array is specified by **rh.recLen**. The **rh.recType** of the **DocInfoListSubcontainerOrAtom** items MUST be one of the following record types: **RT_ProgTags** (section 2.13.24), **RT_NormalViewSetInfo9** (section 2.13.24), **RT_NotesTextViewInfo9** (section 2.13.24), **RT_OutlineViewInfo** (section 2.13.24), **RT_SlideViewInfo** (section 2.13.24), **RT_SorterViewInfo** (section 2.13.24), or **RT_VbaInfo** (section 2.13.24). Each record type MUST NOT occur more than once, except for the **RT_SlideViewInfo** record type, which MUST NOT occur more than twice. If the **RT_SlideViewInfo** record type occurs twice, one occurrence MUST refer to a **SlideViewInfoContainer** record (section 2.4.21.9) and the other occurrence MUST refer to a **NotesViewInfoContainer** record (section 2.4.21.12).

2.4.5 DocInfoListSubContainerOrAtom

Referenced by: DocInfoListContainer

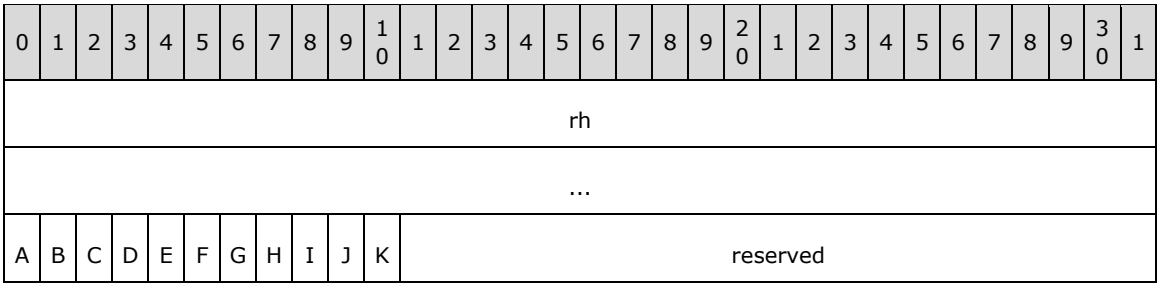
A variable type record whose type and meaning are dictated by the value of **rh.recType**, as specified in the following table.

Value	Meaning
RT_ProgTags (section 2.13.24)	A **DocProgTagsContainer** record (section 2.4.23.1) that specifies programmable tags containing additional document data.
RT_NormalViewSetInfo9 (section 2.13.24)	A **NormalViewSetInfoContainer** record (section 2.4.21.2) that specifies display preferences for a view optimized for the simultaneous display of all presentation slides, a specific presentation slide, and the text of the notes slide associated with that specific presentation slide.
RT_NotesTextViewInfo9 (section 2.13.24)	A **NotesTextViewInfoContainer** record (section 2.4.21.4) that specifies display preferences for a view optimized for the display of the text on the notes slides.
RT_OutlineViewInfo (section 2.13.24)	A **OutlineViewInfoContainer** record (section 2.4.21.6) that specifies display preferences for a view optimized for the display of the text on the presentation slides.
RT_SlideViewInfo (section 2.13.24)	A **SlideViewInfoInstance** record (section 2.4.21.8) that specifies display preferences for a view optimized for the display of presentation slides or notes slides.
RT_SorterViewInfo (section 2.13.24)	A **SorterViewInfoContainer** record (section 2.4.21.13) that specifies display preferences for a view optimized for the simultaneous display of multiple presentation slides.
RT_VbaInfo (section 2.13.24)	A **VBAInfoContainer** record (section 2.4.10) that specifies **VBA** info for the document.

2.4.6 PresAdvisorFlags9Atom

Referenced by: PP9DocBinaryTagExtension

An atom record that specifies which rules to ignore when warning the user about aspects of the document that do not conform to a particular style.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table:

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_PresentationAdvisorFlags9Atom** (section 2.13.24).
rh.recLen	MUST be 0x00000004.

A - fDisableCaseStyleTitleRule (1 bit): A bit that specifies not to warn the user when the letter casing of text in a title placeholder shape does not follow a certain rule.

B - fDisableCaseStyleBodyRule (1 bit): A bit that specifies not to warn the user when the letter casing of text in a body placeholder shape does not follow a certain rule.

C - fDisableEndPunctuationTitleRule (1 bit): A bit that specifies not to warn the user when the ending punctuation of text in a title placeholder shape does not follow a certain rule.

D - fDisableEndPunctuationBodyRule (1 bit): A bit that specifies not to warn the user when the ending punctuation of text in a body placeholder shape does not follow a certain rule.

E - fDisableTooManyBulletsRule (1 bit): A bit that specifies not to warn the user when too many bullets are used.

F - fDisableFontSizeTitleRule (1 bit): A bit that specifies not to warn the user when the font size in a title placeholder shape exceeds a certain size.

G - fDisableFontSizeBodyRule (1 bit): A bit that specifies not to warn the user when the font size in a body placeholder shape exceeds a certain size.

H - fDisableNumberOfLinesTitleRule (1 bit): A bit that specifies not to warn the user when the number of lines of text in a title placeholder shape exceeds a certain quantity.

I - fDisableNumberOfLinesBodyRule (1 bit): A bit that specifies not to warn the user when the number of lines of a paragraph in a body placeholder shape exceeds a certain quantity.

J - fDisableTooManyFontsRule (1 bit): A bit that specifies not to warn the user when the number of different fonts used exceeds a certain quantity.

K - fDisablePrintTip (1 bit): A bit that specifies not to advise the user about printing when they first print the document.

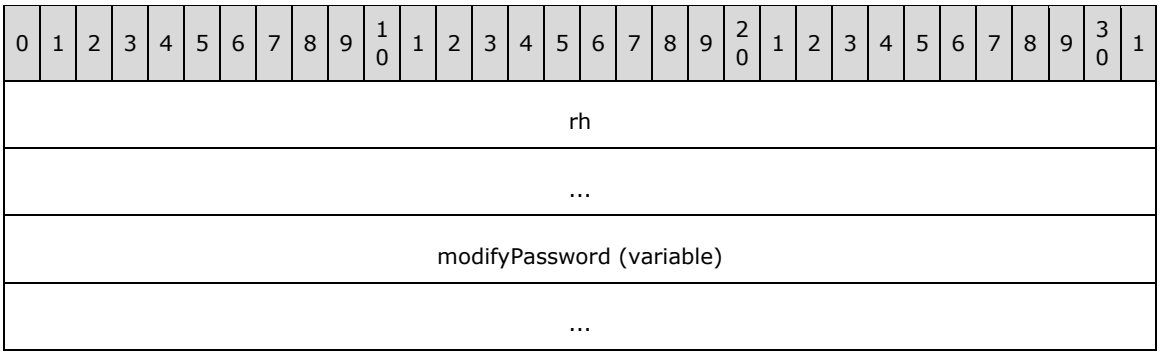
reserved (21 bits): MUST be zero and MUST be ignored.

2.4.7 ModifyPasswordAtom

Referenced by: PP10DocBinaryTagExtension

An atom record that specifies a password used to modify the document.

Files with a modify password MUST be encrypted as specified in [MS-OFFCRYPTO] section 2.4.2.3. An application only grants modify access to the presentation if a user provided password matches the **modifyPassword** field.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

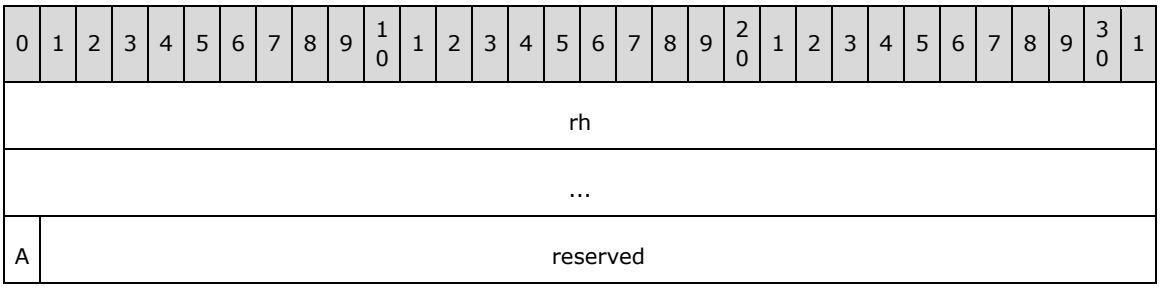
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x003.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 510.

modifyPassword (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies a password used to modify the document. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.8 FilterPrivacyFlags10Atom

Referenced by: PP10DocBinaryTagExtension

An atom record that specifies privacy settings.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_FilterPrivacyFlags10Atom** (section 2.13.24).
rh.recLen	MUST be 0x00000004.

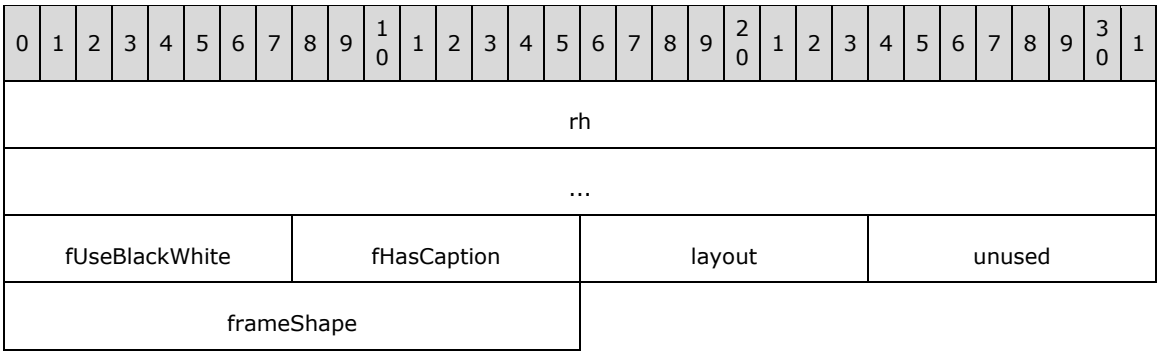
A - fRemovePII (1 bit): A bit that specifies whether personally identifiable information is removed when saving the document.

reserved (31 bits): MUST be zero and MUST be ignored.

2.4.9 PhotoAlbumInfo10Atom

Referenced by: PP10DocBinaryTagExtension

An atom record that specifies information about how to display a presentation as a photo album.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_PhotoAlbumInfo10Atom** (section 2.13.24).
rh.recLen	MUST be 0x00000006.

fUseBlackWhite (1 byte): A **bool1** (section 2.2.2) that specifies a user preference for whether to display all pictures in grayscale graphics.

fHasCaption (1 byte): A **bool1** that specifies a user preference for whether a text caption exists beneath each picture in the album.

layout (1 byte): A **PhotoAlbumLayoutEnum** enumeration (section 2.13.20) that specifies a user preference for the layout of the photos in this presentation.

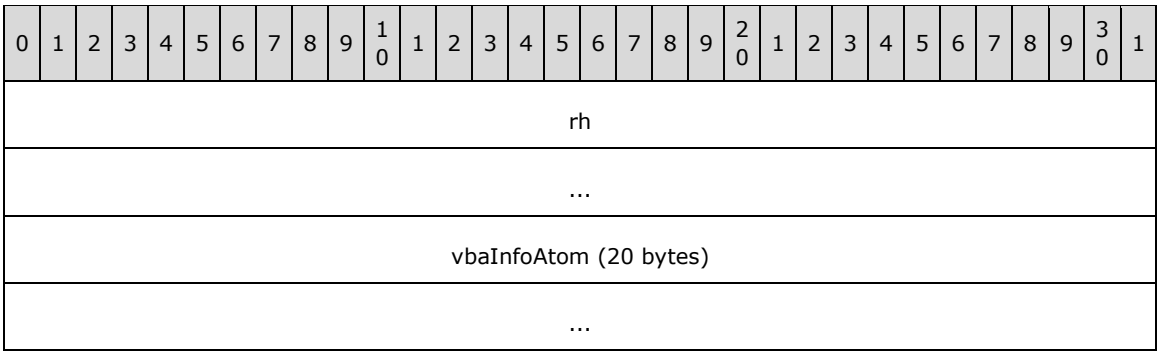
unused (1 byte): Undefined and MUST be ignored.

frameShape (2 bytes): A **PhotoAlbumFrameShapeEnum** enumeration that specifies a user preference for the shape of the frame around each photo.

2.4.10 VBAInfoContainer

Referenced by: DocInfoListSubContainerOrAtom

A container record that specifies VBA information for the document.



...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be **RT_VbaInfo** (section 2.13.24).
rh.recLen	MUST be 0x00000014.

vbaInfoAtom (20 bytes): A VBAInfoAtom record that specifies VBA information for this document.

2.4.11 VBAInfoAtom

Referenced by: VBAInfoContainer

An atom record that specifies a reference to the VBA project storage.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
persistIdRef																															
fHasMacros																															
version																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x2.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_VbaInfoAtom.
rh.recLen	MUST be 0x0000000C.

persistIdRef (4 bytes): A **PersistIdRef** (section 2.2.21) that specifies the value to look up in the persist object directory to find the offset of a **VbaProjectStg** record (section 2.10.40).

fHasMacros (4 bytes): An **unsigned integer** that specifies whether the VBA project storage contains data. It MUST be a value from the following table.

Value	Meaning
0x00000000	The VBA storage is empty.

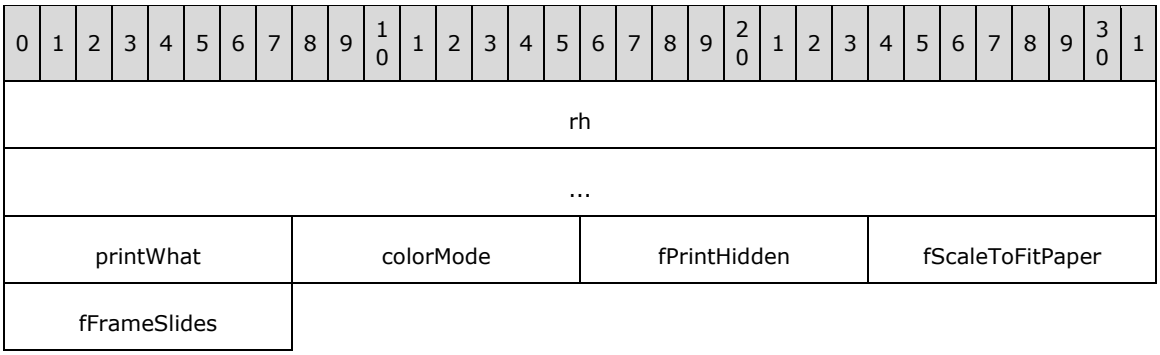
Value	Meaning
0x00000001	The VBA storage contains data.

version (4 bytes): An unsigned integer that specifies the VBA runtime version that generated the VBA project storage. It MUST be 0x00000002.

2.4.12 PrintOptionsAtom

Referenced by: DocumentContainer

An atom record that specifies user preferences for printing the document.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_PrintOptionsAtom.
rh.recLen	MUST be 0x00000005.

printWhat (1 byte): A **PrintWhatEnum** enumeration that specifies what is printed.

colorMode (1 byte): A **ColorModeEnum** enumeration that specifies how colors are printed.

fPrintHidden (1 byte): A **bool1** (section 2.2.2) that specifies whether hidden slides are printed.

fScaleToFitPaper (1 byte): A **bool1** that specifies whether the slide is scaled as large as possible to fit the printable area of the page and maintain its aspect ratio.

fFrameSlides (1 byte): A **bool1** that specifies whether a border is drawn around each slide.

2.4.13 EndDocumentAtom

Referenced by: DocumentContainer

An atom record that specifies the end of information for the document inside a **DocumentContainer** record (section 2.4.1).

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_EndDocumentAtom.
rh.recLen	MUST be 0x00000000.

2.4.14 Slide List Types

2.4.14.1 MasterListWithTextContainer

Referenced by: DocumentContainer

A container record that specifies a list of references to main master slides and title master slides.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
rgMasterPersistAtom (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

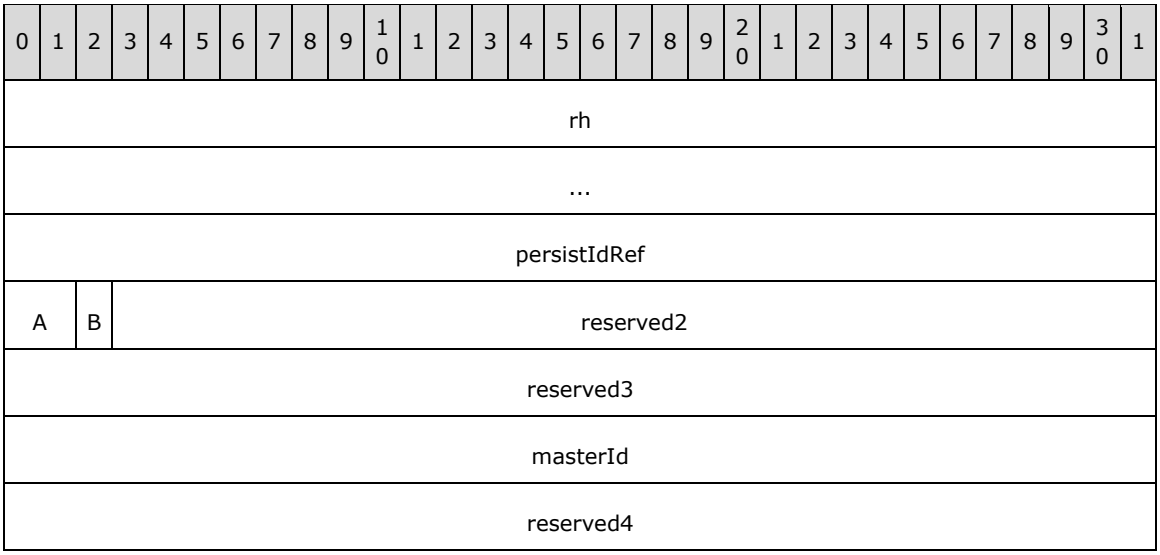
Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be RT_SlideListWithText.

rgMasterPersistAtom (variable): An array of **MasterPersistAtom** records (section 2.4.14.2) that specifies references to the main master slides and title master slides. The length, in bytes, of the array is specified by **rh.recLen**.

2.4.14.2 MasterPersistAtom

Referenced by: MasterListWithTextContainer

An atom record that specifies a reference to a main master slide or title master slide.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_SlidePersistAtom.
rh.recLen	MUST be 0x00000014.

persistIdRef (4 bytes): A **PersistIdRef** (section 2.2.21) that specifies the value to look up in the persist object directory to find the offset of the **MainMasterContainer** record (section 2.5.3) for a main master slide or a **SlideContainer** record (section 2.5.1) for a title master slide.

A - reserved1 (2 bits): MUST be zero and MUST be ignored.

B - fNonOutlineData (1 bit): A bit that specifies whether the main master slide or title master slide specified by the **persistIdRef** field contains data other than text in a placeholder shape.

reserved2 (29 bits): MUST be zero and MUST be ignored.

reserved3 (4 bytes): MUST be zero and MUST be ignored.

masterId (4 bytes): A MasterId that specifies the identifier for the main master slide or title master slide specified by the **persistIdRef** field.

reserved4 (4 bytes): MUST be zero and MUST be ignored.

2.4.14.3 SlideListWithTextContainer

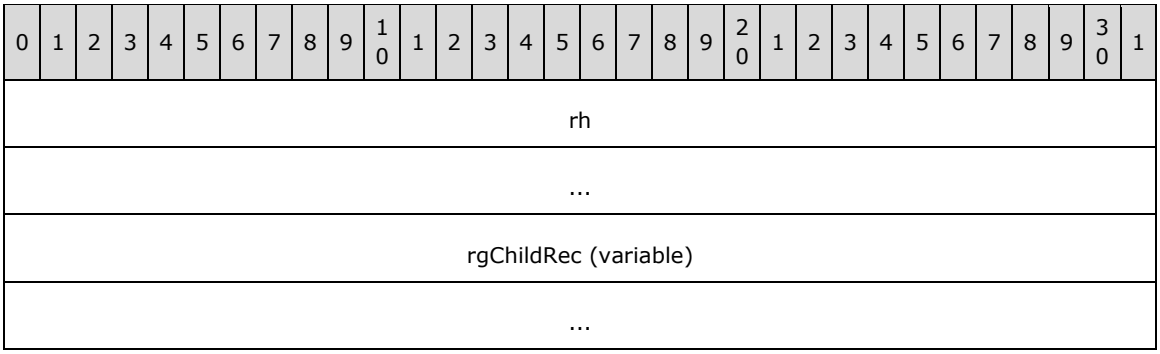
Referenced by: DocumentContainer

A container record that specifies a list of references to presentation slides and text-related records for text contained within those presentation slides.

Each **SlidePersistAtom** record (section 2.4.14.5) in this list references a **SlideContainer** record (section 2.5.1) as specified by the **persistIdRef** field of the **SlidePersistAtom** record. Let the *corresponding slide* be the **SlideContainer** record so specified.

Let the *corresponding text placeholder list* be specified by the sequence of items in the **slideAtom.rgPlaceholderTypes** array of the *corresponding slide* with one of the following values: PT_MasterTitle, PT_MasterBody, PT_MasterCenterTitle, PT_MasterSubTitle, PT_Title, PT_Body, PT_CenterTitle, PT_SubTitle, PT_VerticalTitle, or PT_VerticalBody.

The i^{th} TextHeaderAtom record that follows a **SlidePersistAtom** record specifies the text of a shape that corresponds to the i^{th} item in the *corresponding text* placeholder list.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_SlideListWithText.

rgChildRec (variable): An array of SlideListWithTextSubContainerOrAtom records that specifies the references to presentation slides and text contained within those presentation slides. The sequence of the **rh.recType** fields of array items MUST be a valid **SlideListWithTextRecordList** as specified in the following ABNF (specified in [RFC5234]) grammar:

```
SlideListWithTextRecordList = 1*SlideRecordList
SlideRecordList             = RT_SlidePersistAtom *8(RT_TextHeaderAtom TextCharsOrBytesRecord
StyleTextPropRecord MetaCharRecordList TextBookmarkRecordList TextSpecialInfoRecord
InteractiveRecordList)
TextCharsOrBytesRecord      = *1(RT_TextCharsAtom / RT_TextBytesAtom)
StyleTextPropRecord         = *1RT_StyleTextPropAtom
MetaCharRecordList          = *(RT_SlideNumberMetaCharAtom / RT_DateTimeMetaCharAtom /
RT_GenericDateMetaCharAtom / RT_HeaderMetaCharAtom / RT_FooterMetaCharAtom /
RT_RtfDateTimeMetaCharAtom)
TextBookmarkRecordList      = *RT_TextBookmarkAtom
TextSpecialInfoRecord       = *1RT_TextSpecialInfoAtom
InteractiveRecordList       = *(RT_InteractiveInfo RT_TextInteractiveInfoAtom)
```

2.4.14.4 SlideListWithTextSubContainerOrAtom

Referenced by: SlideListWithTextContainer

A variable type record whose type and meaning are dictated by the value of **rh.recType**, as specified in the following table.

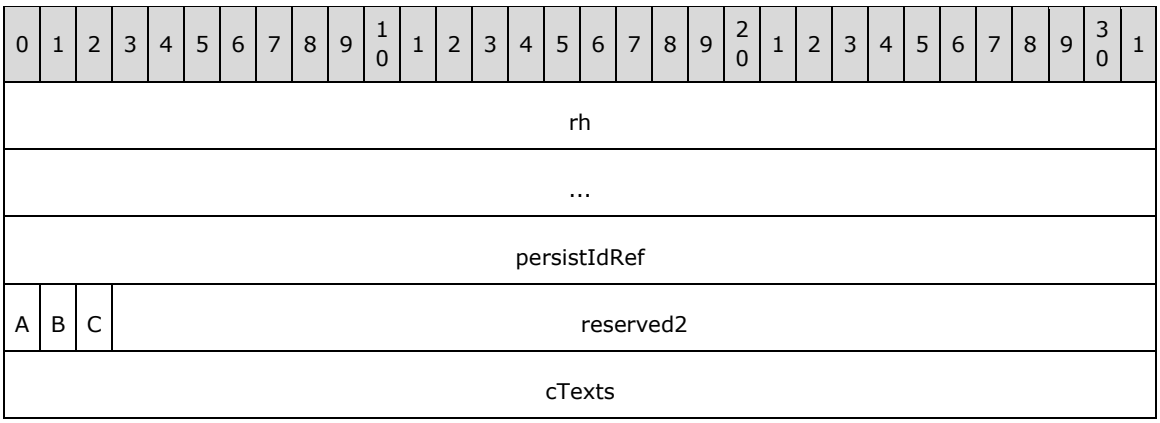
Value	Meaning
RT_SlidePersistAtom	A **SlidePersistAtom** record (section 2.4.14.5) that specifies a reference to a presentation slide.
RT_TextHeaderAtom	A TextHeaderAtom record that specifies the type of a body of text.
RT_TextCharsAtom	A TextCharsAtom record that specifies text characters.
RT_TextBytesAtom	A TextBytesAtom record that specifies text characters.
RT_StyleTextPropAtom	A StyleTextPropAtom record that specifies text character and paragraph properties.
RT_SlideNumberMetaCharAtom	A SlideNumberMCAtom record that specifies a slide number metacharacter.
RT_DateTimeMetaCharAtom	A DateTimeMCAtom record that specifies a **datetime** metacharacter.
RT_GenericDateMetaCharAtom	A GenericDateMCAtom record that specifies a **datetime** metacharacter.
RT_HeaderMetaCharAtom	A HeaderMCAtom record that specifies a header metacharacter.
RT_FooterMetaCharAtom	A FooterMCAtom record that specifies a footer metacharacter.
RT_RtfDateTimeMetaCharAtom	A RTFDateTimeMCAtom record that specifies an RTF **datetime** metacharacter.
RT_TextBookmarkAtom	A TextBookmarkAtom record that specifies a text bookmark.
RT_TextSpecialInfoAtom	A TextSpecialInfoAtom record that specifies additional text properties.
RT_InteractiveInfo	An InteractiveInfoInstance record that specifies text interactive information.
RT_TextInteractiveInfoAtom	A TextInteractiveInfoInstance record that specifies the anchor for text interactive information.

2.4.14.5 SlidePersistAtom

Referenced by: SlideListWithTextSubContainerOrAtom

An atom record that specifies a reference to a presentation slide.

Let the *corresponding slide* be as specified by the **persistIdRef** field.



slideId
reserved3

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_SlidePersistAtom.
rh.recLen	MUST be 0x00000014.

persistIdRef (4 bytes): A **PersistIdRef** (section 2.2.21) that specifies the value to look up in the persist object directory to find the offset of the **SlideContainer** record (section 2.5.1) for a presentation slide.

A - reserved1 (1 bit): MUST be zero and MUST be ignored.

B - fShouldCollapse (1 bit): A **bit** that specifies whether the *corresponding slide* is collapsed.

C - fNonOutlineData (1 bit): A **bit** that specifies whether the *corresponding slide* contains data other than text in a placeholder shape.

reserved2 (29 bits): MUST be zero and MUST be ignored.

cTexts (4 bytes): A **signed integer** that specifies the number of text placeholder shapes on the *corresponding slide*. It MUST be greater than or equal to 0x00000000. It SHOULD be less than or equal to 0x00000005 and MUST be less than or equal to 0x00000008.

slideId (4 bytes): A SlideId that specifies the identifier for the *corresponding slide*.

reserved3 (4 bytes): MUST be zero and MUST be ignored.

2.4.14.6 NotesListWithTextContainer

Referenced by: DocumentContainer

A container record that specifies a list of references to notes slides.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
rgNotesPersistAtom (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

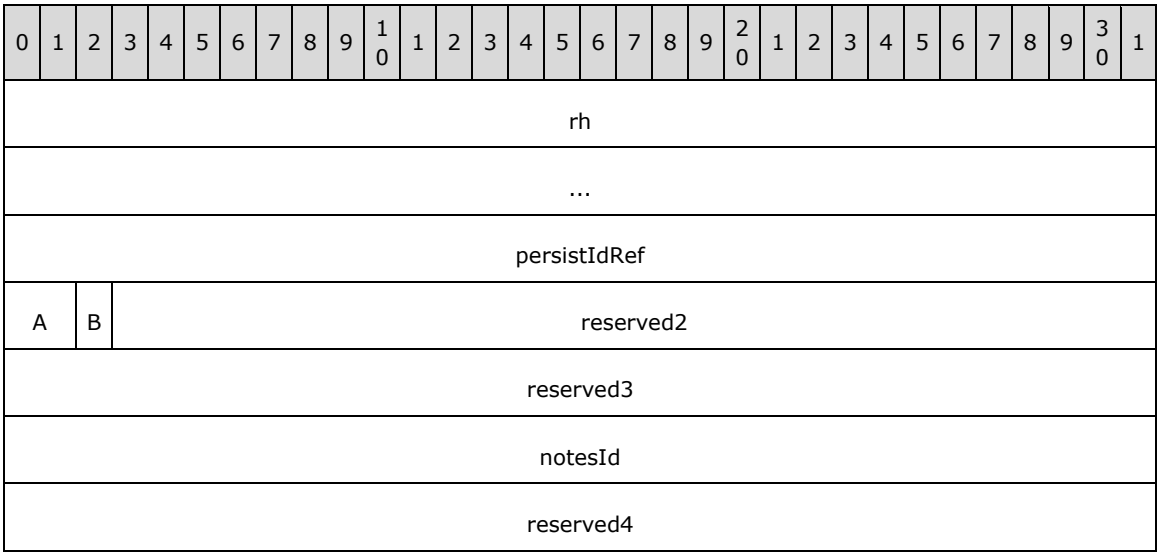
Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x002.
rh.recType	MUST be RT_SlideListWithText.

rgNotesPersistAtom (variable): An array of **NotesPersistAtom** records (section 2.4.14.7) that specifies references to notes slides. The length, in bytes, of the array is specified by **rh.recLen**.

2.4.14.7 NotesPersistAtom

Referenced by: NotesListWithTextContainer

An atom record that specifies a reference to a notes slide.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_SlidePersistAtom.
rh.recLen	MUST be 0x00000014.

persistIdRef (4 bytes): A **PersistIdRef** (section 2.2.21) that specifies the value to look up in the persist object directory to find the offset of the **NotesContainer** record (section 2.5.6) for a notes slide.

A - reserved1 (2 bits): MUST be zero and MUST be ignored.

B - fNonOutlineData (1 bit): A bit that specifies whether the notes slide specified by the **persistIdRef** field contains data other than text in a placeholder shape.

reserved2 (29 bits): MUST be zero and MUST be ignored.

reserved3 (4 bytes): MUST be zero and MUST be ignored.

notesId (4 bytes): A NotesId that specifies the identifier for the notes slide specified by the **persistIdRef** field.

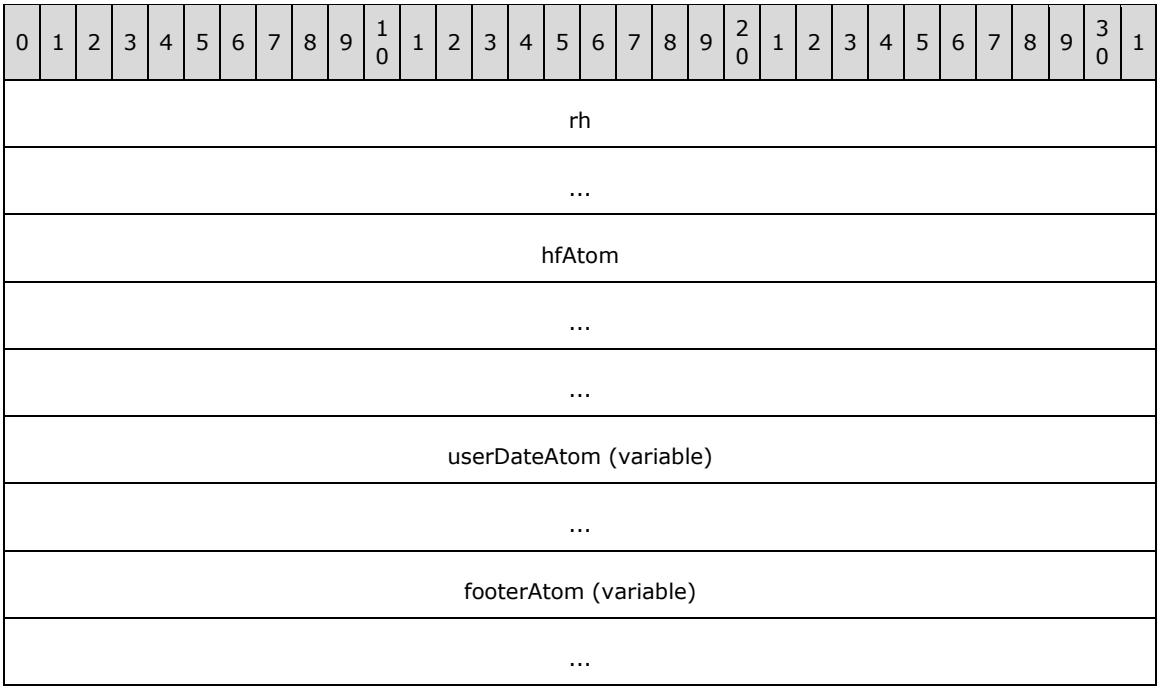
reserved4 (4 bytes): MUST be zero and MUST be ignored.

2.4.15 Header/Footer Types

2.4.15.1 SlideHeadersFootersContainer

Referenced by: DocumentContainer

A container record that specifies information about the footers on a presentation slide.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x003.
rh.recType	MUST be RT_HeadersFooters.

hfAtom (12 bytes): A HeadersFootersAtom record that specifies the options for displaying the footers. The **hfAtom.fHasHeader** sub-field MUST be ignored.

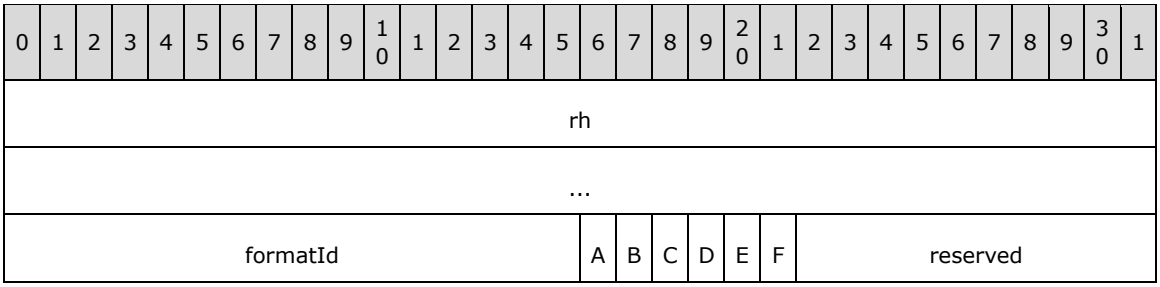
userDateAtom (variable): An optional UserDateAtom record that specifies the custom date to be used in the date field.

footerAtom (variable): An optional FooterAtom record that specifies the content of the footer.

2.4.15.2 HeadersFootersAtom

Referenced by: NotesHeadersFootersContainer, PerSlideHeadersFootersContainer, SlideHeadersFootersContainer

An atom record that specifies options for displaying headers and footers on a presentation slide or notes slide.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_HeadersFootersAtom.
rh.recLen	MUST be 0x00000004.

formatId (2 bytes): A signed integer that specifies the format identifier to be used to style the date and time. It MUST be greater than or equal to 0x0000 and less than or equal to 0x000D. It SHOULD<12> be less than or equal to 0x000C. This value is converted into a string as specified by the **index** field of the DateTimeMCAtom record. It MUST be ignored unless **fHasTodayDate** is **TRUE**.

A - fHasDate (1 bit): A bit that specifies whether the date is displayed in the footer.

B - fHasTodayDate (1 bit): A bit that specifies whether the current **datetime** is used for displaying the **datetime**.

C - fHasUserDate (1 bit): A bit that specifies whether the date specified in UserDateAtom record is used for displaying the **datetime**.

D - fHasSlideNumber (1 bit): A bit that specifies whether the slide number is displayed in the footer.

E - fHasHeader (1 bit): A bit that specifies whether the header text specified by HeaderAtom record is displayed.

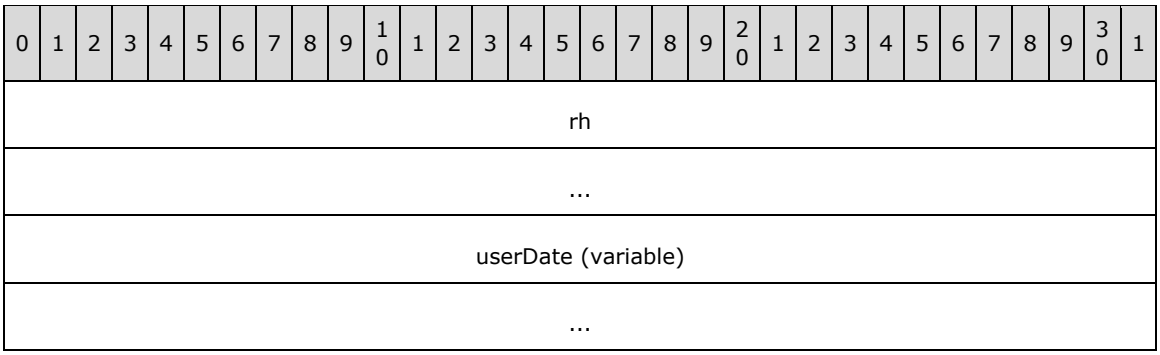
F - fHasFooter (1 bit): A bit that specifies whether the footer text specified by FooterAtom record is displayed.

reserved (10 bits): MUST be zero and MUST be ignored.

2.4.15.3 UserDateAtom

Referenced by: NotesHeadersFootersContainer, PerSlideHeadersFootersContainer, SlideHeadersFootersContainer

An atom record that specifies the custom date for use in headers and footers.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

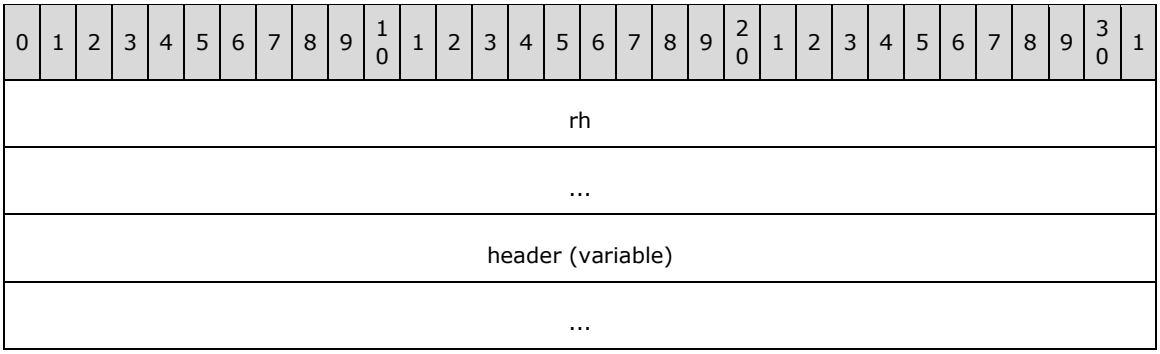
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 510.

userDate (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the custom date. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.15.4 HeaderAtom

Referenced by: NotesHeadersFootersContainer

An atom record that specifies text to be used in a header on a handout slide or notes slide.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

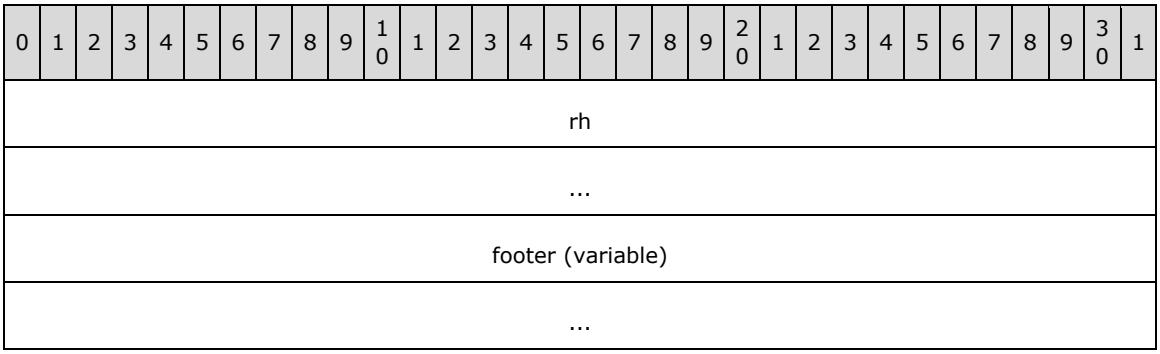
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number.

header (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the text for the header. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.15.5 FooterAtom

Referenced by: NotesHeadersFootersContainer, PerSlideHeadersFootersContainer, SlideHeadersFootersContainer

An atom record that specifies text to be used in a footer on a presentation slide, handout slide or notes slide.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

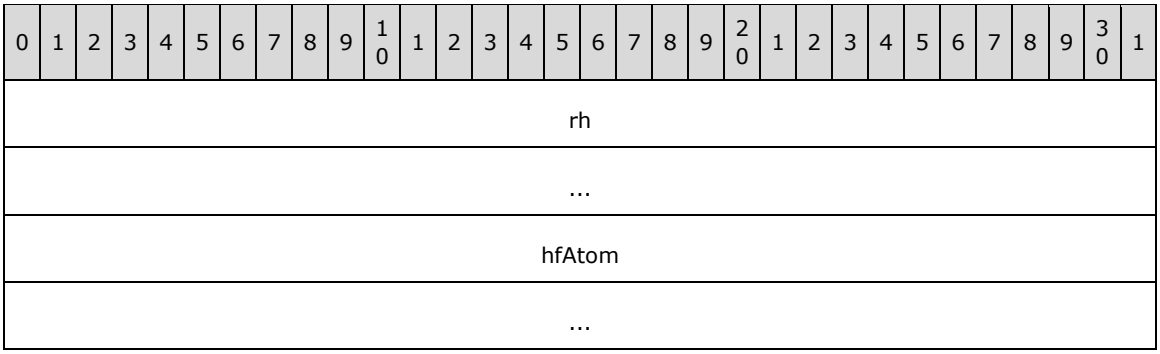
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x002.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number.

footer (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the text for the footer. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.15.6 NotesHeadersFootersContainer

Referenced by: DocumentContainer

A container record that specifies information about the headers and footers on a notes slide.



...
userDateAtom (variable)
...
headerAtom (variable)
...
footerAtom (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x004.
rh.recType	MUST be RT_HeadersFooters.

hfAtom (12 bytes): A HeadersFootersAtom record that specifies the options for displaying the headers and footers.

userDateAtom (variable): An optional UserDateAtom record that specifies the custom date to be used in a HeadersFootersAtom record.

headerAtom (variable): An optional HeaderAtom record that specifies the content of the header.

footerAtom (variable): An optional FooterAtom record that specifies the content of the footer.

2.4.16 Sound Types

2.4.16.1 SoundCollectionContainer

Referenced by: DocumentContainer

A container record that specifies all embedded sounds in the document.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
soundCollectionAtom																															
...																															

...
rgSoundContainer (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x005.
rh.recType	MUST be RT_SoundCollection.

soundCollectionAtom (12 bytes): A SoundCollectionAtom record that specifies the seed for creating new **sound identifiers** for sounds in this collection.

rgSoundContainer (variable): An array of **SoundContainer** records (section 2.4.16.3) that specifies the embedded sounds. The length, in bytes, of the array is specified by the following formula:

```
rh.recLen - 12.
```

2.4.16.2 SoundCollectionAtom

Referenced by: SoundCollectionContainer

An atom record that specifies the seed for creating new sound identifiers for sounds in the **SoundCollectionContainer** record (section 2.4.16.1).

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
soundIdSeed																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

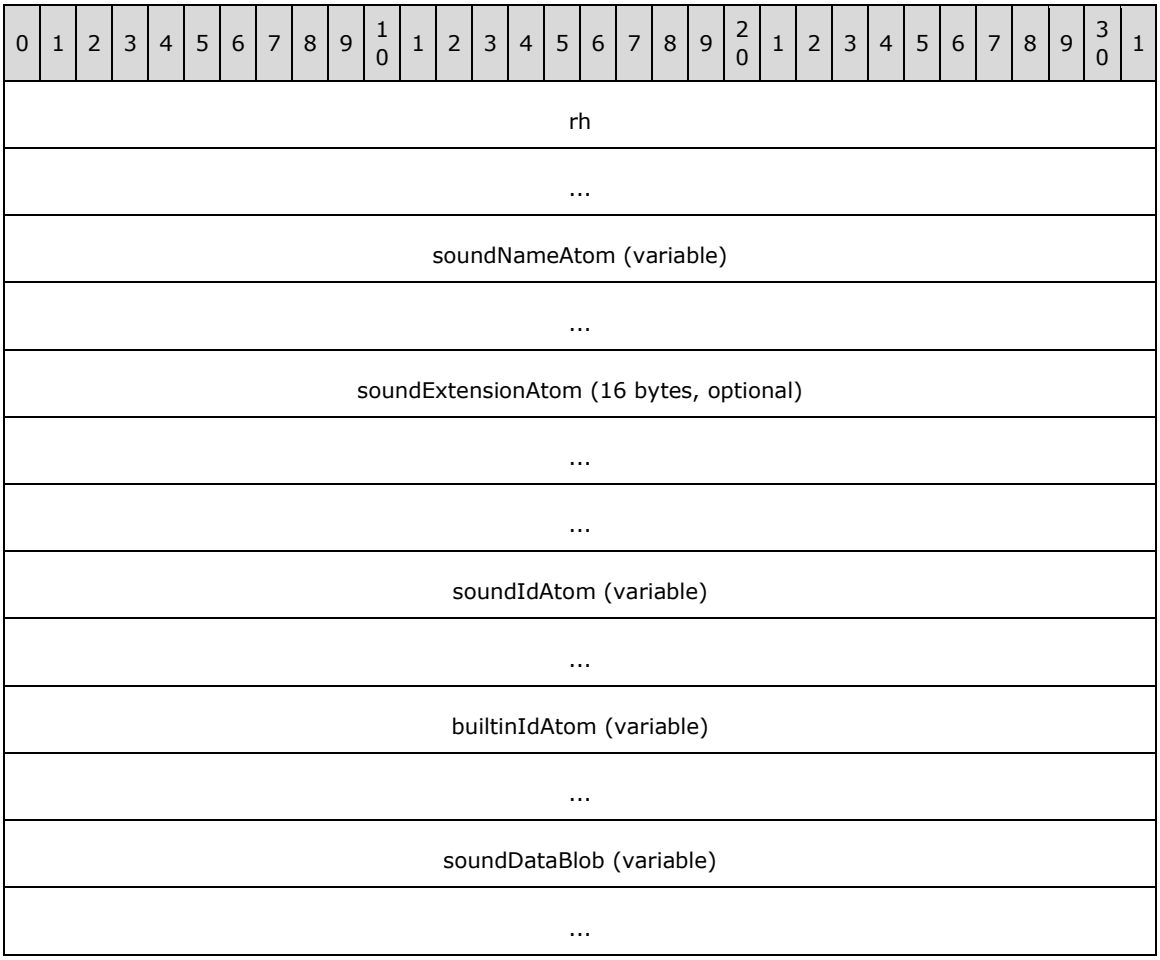
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_SoundCollectionAtom.
rh.recLen	MUST be 0x00000004.

soundIdSeed (4 bytes): A signed integer that specifies the seed for creating a new sound identifier. It MUST be greater than 0x00000000 and greater than or equal to all sound identifiers specified by the SoundIdAtom records.

2.4.16.3 SoundContainer

Referenced by: AnimationInfoContainer, SoundCollectionContainer

A container record that specifies information about an embedded sound.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_Sound.

soundNameAtom (variable): A SoundNameAtom record that specifies the name of the sound.

soundExtensionAtom (16 bytes): An optional SoundExtensionAtom record that specifies the format of the audio data.

soundIdAtom (variable): A SoundIdAtom record that specifies the sound identifier for the sound.

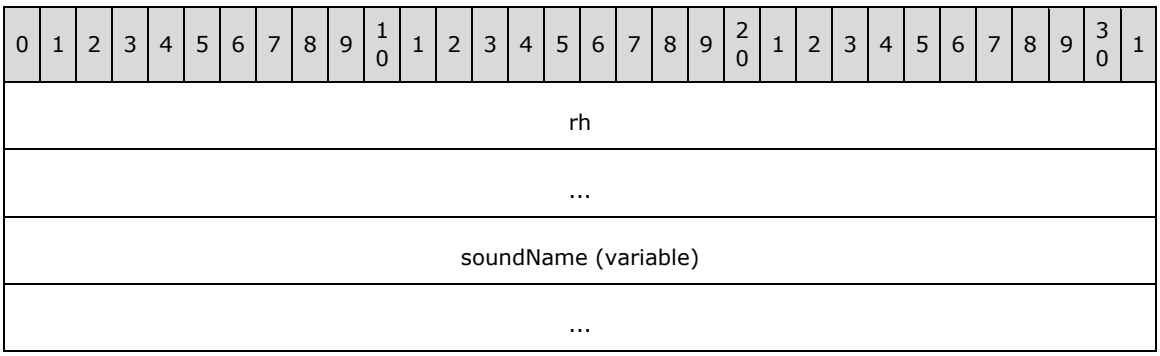
builtinIdAtom (variable): An optional SoundBuiltinIdAtom record that specifies an identifier that describes the sound.

soundDataBlob (variable): A SoundDataBlob record that specifies the audio data for the sound.

2.4.16.4 SoundNameAtom

Referenced by: SoundContainer

An atom record that specifies the name of a sound.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

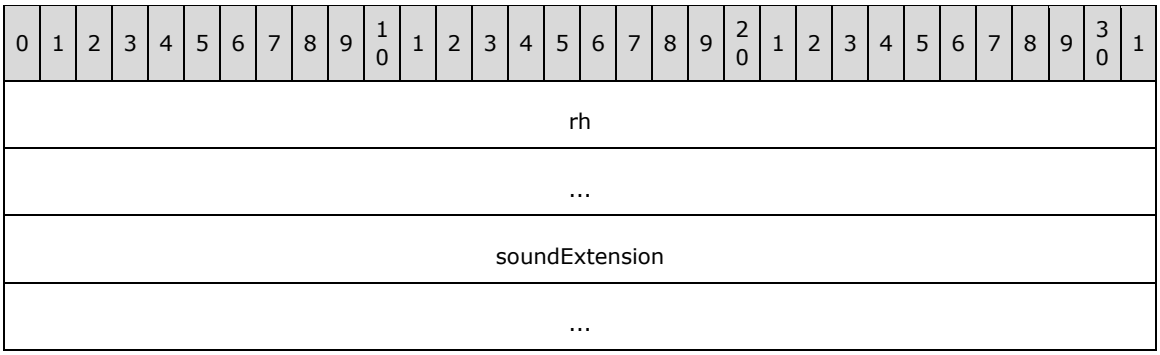
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number.

soundName (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the name of a sound. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.16.5 SoundExtensionAtom

Referenced by: SoundContainer

An atom record that specifies the format of the audio data for a sound.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be 0x00000008.

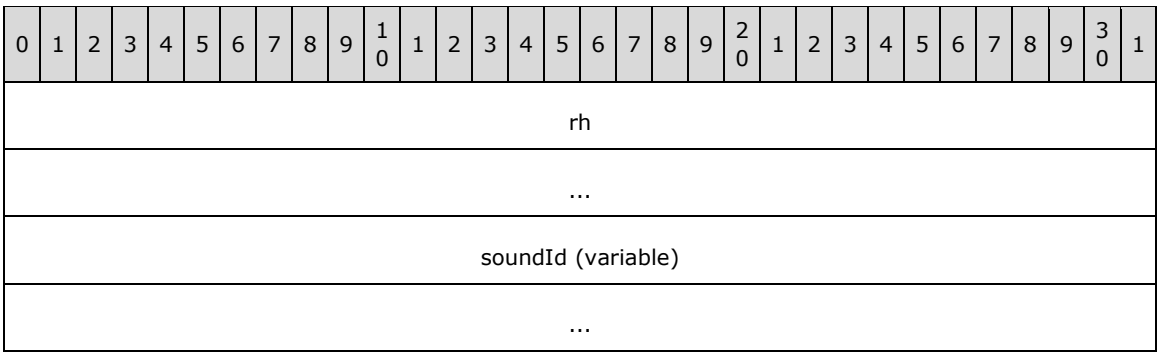
soundExtension (8 bytes): A **UTF-16 Unicode** [RFC2781] **string** that specifies the format of the audio data for a sound. It SHOULD<13> be a value from the following table.

Value (case-insensitive)	Meaning
.wav	The format is **WAV**.
wave	The format is WAV.
.aif	The format is **Audio Interchange File Format (AIFF)**.
aiff	The format is AIFF.

2.4.16.6 SoundIdAtom

Referenced by: SoundContainer

An atom record that specifies the sound identifier for a sound.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x002.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number.

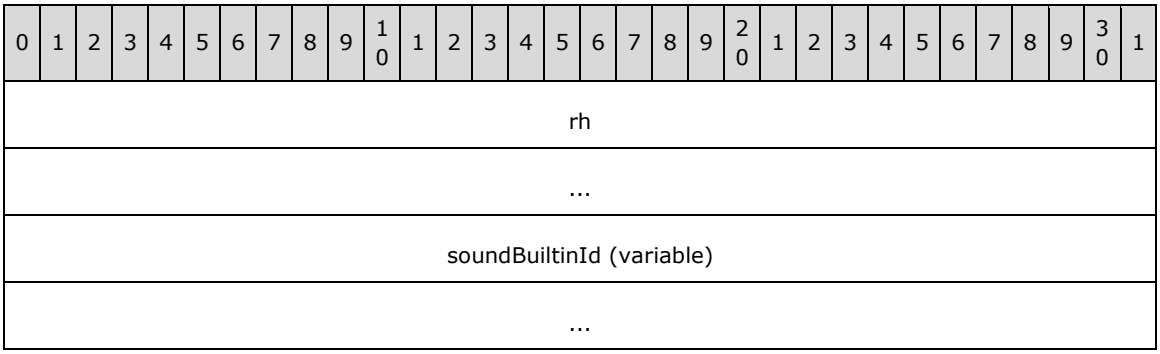
soundId (variable): A UTF-16 Unicode [RFC2781] string representation of the base-10 form of an integer value that specifies the sound identifier for a sound. The integer value MUST be greater than zero, less than or equal to the seed specified by the SoundCollectionAtom record and unique

within the **SoundCollectionContainer** record (section 2.4.16.1). The length, in bytes, of the field is specified by **rh.recLen**.

2.4.16.7 SoundBuiltinIdAtom

Referenced by: SoundContainer

An atom record that specifies a description of a sound.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x003.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number.

soundBuiltinId (variable): A UTF-16 Unicode [RFC2781] string representation of the base-10 form of an integer value that specifies a description of a sound. It MUST be a value from the following table.

Value	Meaning
100	Cash Register
101	Typewriter
102	Screeching Brakes
103	Whoosh
104	Laser
105	Camera
106	Chime
107	Clapping
108	Applause
109	Drive By
110	Drum Roll
111	Explosion
112	Breaking Glass
113	Gunshot
114	Slide Projector
115	Ricochet

Value	Meaning
116	Arrow
117	Bomb
118	Breeze
119	Click
120	Coin
121	Hammer
122	Push
123	Suction
124	Voltage
125	Wind

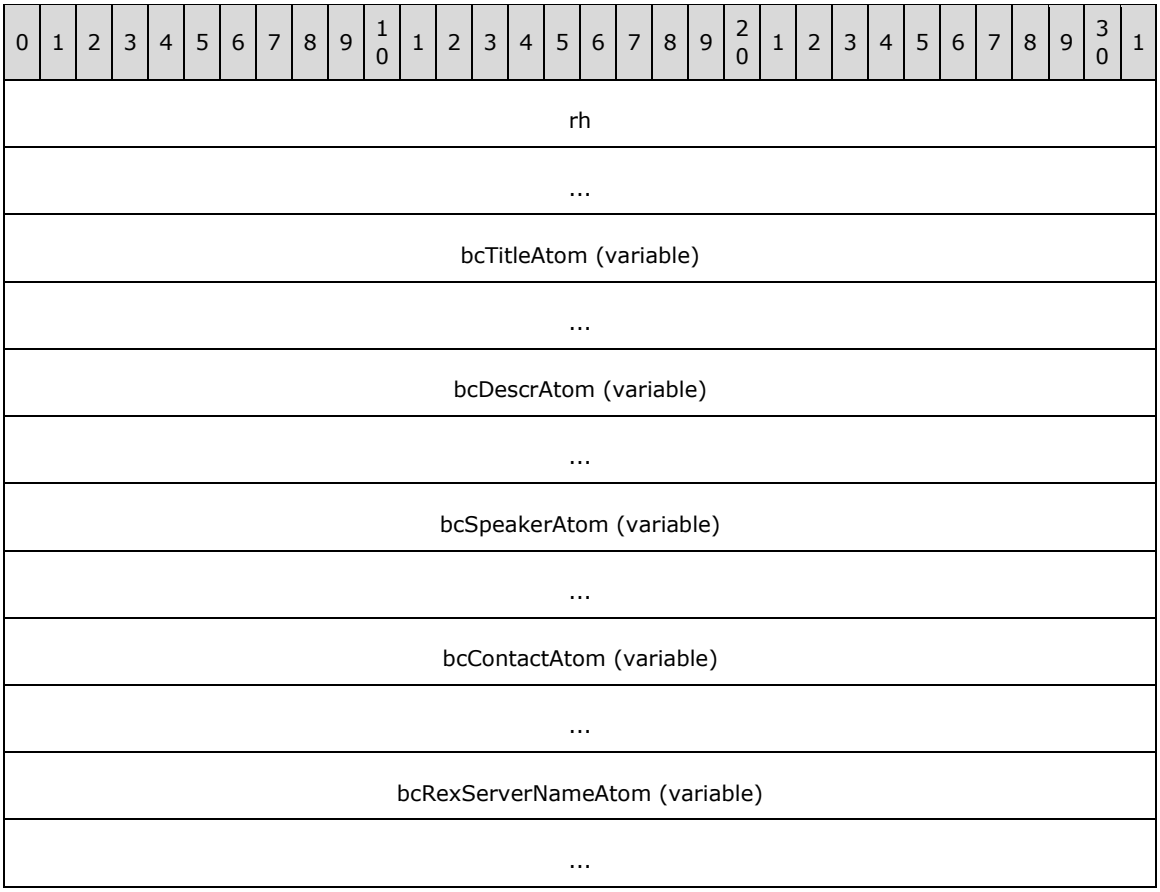
The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17 Broadcast Types

2.4.17.1 BroadcastDocInfo9Container

Referenced by: PP9DocBinaryTagExtension

A container record that specifies settings for a **presentation broadcast**. It SHOULD<14> be ignored. Some settings refer to NetShow; see [MSFT-UMWNSNS] for more information.



bcEmailAddressAtom (variable)
...
bcEmailNameAtom (variable)
...
bcChatUrlAtom (variable)
...
bcArchiveDirAtom (variable)
...
bcNSFilesBaseDirAtom (variable)
...
bcNSFilesDirAtom (variable)
...
bcNSServerNameAtom (variable)
...
bcPptFilesBaseDirAtom (variable)
...
bcPptFilesDirAtom (variable)
...
bcPptFilesBaseUrlAtom (variable)
...
bcUserNameAtom (variable)
...
bcBroadcastDateTimeAtom (variable)
...
bcPresentationNameAtom (variable)

...
bcAsdFileNameAtom (variable)
...
bcEntryIdAtom (variable)
...
bcDocInfoAtom (42 bytes)
...
...
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table:

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_BroadcastDocInfo9.

bcTitleAtom (variable): An optional BCTitleAtom record that specifies the title.

bcDescrAtom (variable): An optional BCDescriptionAtom record that specifies the description.

bcSpeakerAtom (variable): An optional BCSpeakerAtom record that specifies the name of the speaker.

bcContactAtom (variable): An optional BCContactAtom record that specifies the name of the contact person.

bcRexServerNameAtom (variable): An optional BCRexServerNameAtom record that specifies the name of the remote computer to which a camera or microphone is connected to record the video or audio. It MUST exist if the **fCameraRemote** field of the BroadcastDocInfoAtom record is set to **TRUE**.

bcEmailAddressAtom (variable): An optional BCEmailAddressAtom record that specifies the e-mail address for audience feedback.

bcEmailNameAtom (variable): An optional BCEmailNameAtom record that specifies the e-mail name for audience feedback. It MUST exist if **bcDocInfoAtom.fCanEmail** is **TRUE**.

bcChatUrlAtom (variable): An optional BCChatUrlAtom record that specifies the URL of a chat server**.**

bcArchiveDirAtom (variable): An optional BCArchiveDirAtom record that specifies the directory location to archive this presentation broadcast.

bcNSFilesBaseDirAtom (variable): An optional BCNetShowFilesBaseDirAtom record that specifies the UNC base directory to store presentation broadcast files for NetShow.

bcNSFilesDirAtom (variable): An optional BCNetShowFilesDirAtom record that specifies the UNC directory location to store presentation broadcast files for NetShow. It MUST exist if **bcDocInfoAtom.fUseNetShow** is **TRUE**.

bcNSServerNameAtom (variable): An optional BCNetShowServerNameAtom record that specifies the name of the NetShow server. It MUST exist if **bcDocInfoAtom.fUseNetShow** is **TRUE**.

bcPptFilesBaseDirAtom (variable): A BCPptFilesBaseDirAtom record that specifies the path to the UNC base directory to store presentation broadcast files.

bcPptFilesDirAtom (variable): A BCPptFilesDirAtom record that specifies the path to the UNC directory to store presentation broadcast files.

bcPptFilesBaseUrlAtom (variable): A BCPptFilesBaseUrlAtom record that specifies the UNC or HTTP location of the directory specified in **bcPptFilesDirAtom**.

bcUserNameAtom (variable): A BCUserNameAtom record that specifies the name of the user who scheduled the presentation broadcast.

bcBroadcastDateTimeAtom (variable): A BCBroadcastDateTimeAtom record that specifies the directory name to create under the base directory specified in **bcPptFilesBaseDirAtom**.

bcPresentationNameAtom (variable): A BCPresentationNameAtom record that specifies the name of the presentation.

bcAsdFileNameAtom (variable): A BCAsdFileNameAtom record that specifies the location of an ASD file. The ASD file is the description file for an Advanced Systems Format (ASF) file, described in [ASF], used to stream audio and video content.

bcEntryIdAtom (variable): An optional BCEntryIDAtom record that specifies the identifier for a calendar item to associate with this presentation broadcast.

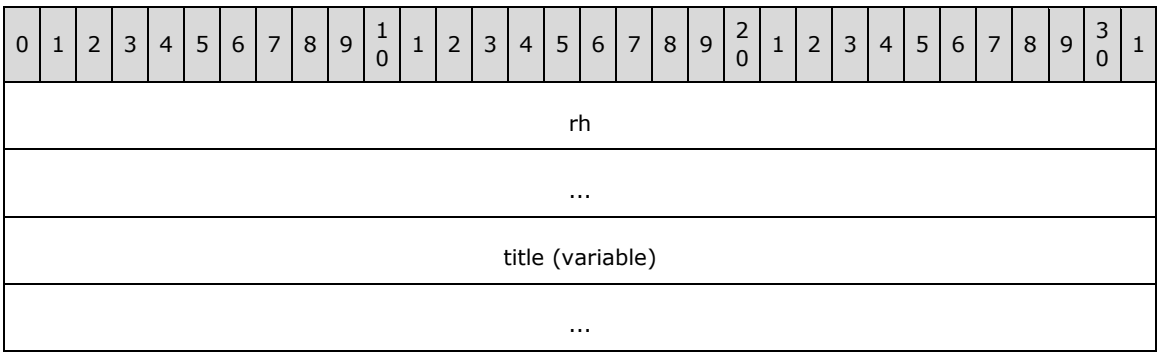
bcDocInfoAtom (42 bytes): A BroadcastDocInfoAtom record that specifies properties of a presentation broadcast.

2.4.17.2 BCTitleAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the title of a presentation broadcast.

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCTitleAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 510.

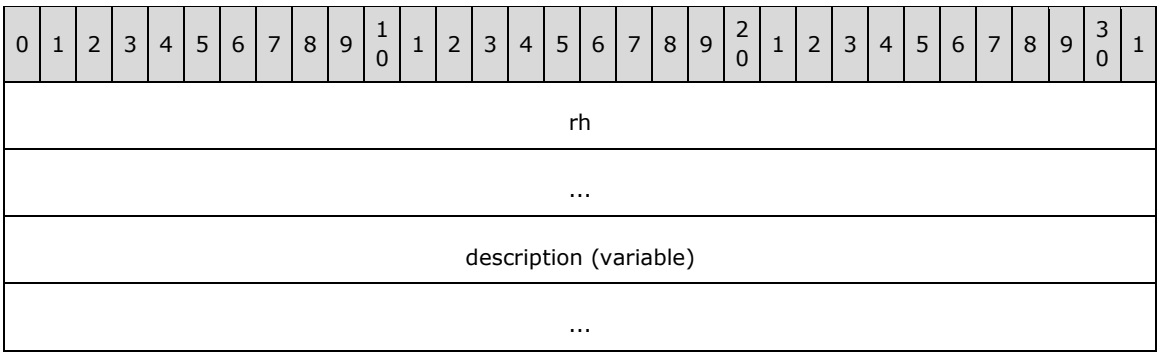
title (variable): A UnicodeString that specifies the title of the *corresponding presentation broadcast*. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.3 BCDescriptionAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the description of a presentation broadcast.

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCDescriptionAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x002.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 2040.

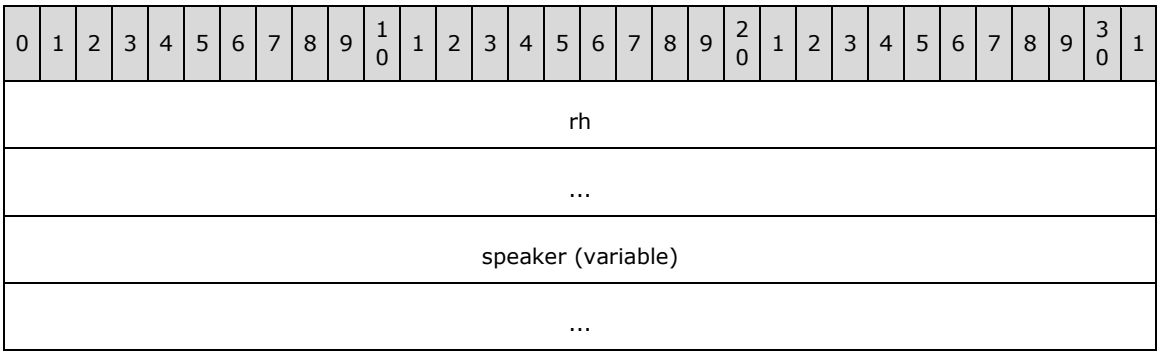
description (variable): A UnicodeString that specifies the description of the corresponding presentation broadcast. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.4 BCSpeakerAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the name of the speaker for a presentation broadcast.

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCSpeakerAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x003.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 510.

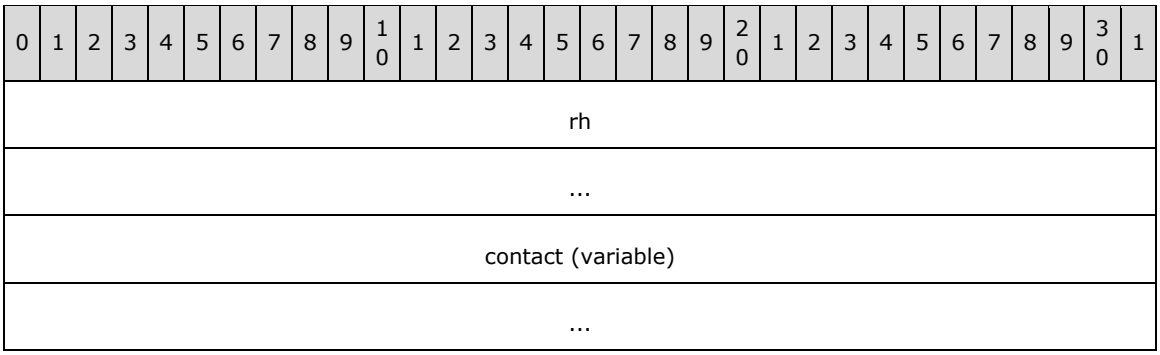
speaker (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the name of the speaker for the *corresponding presentation broadcast*. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.5 BCContactAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the name of the contact person for a presentation broadcast.

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCContactAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x004.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 510.

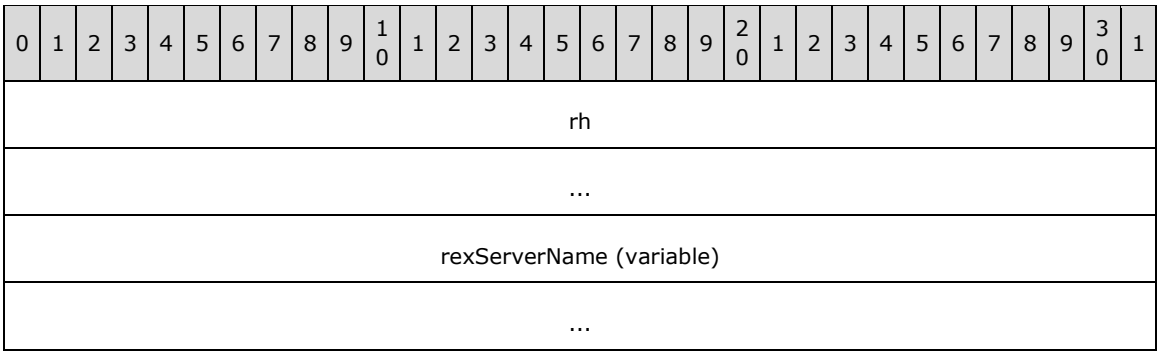
contact (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the name of the contact person for the *corresponding presentation broadcast*. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.6 BCRexServerNameAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the name of the remote computer to which a camera or microphone is connected to record the video or audio of a presentation broadcast.

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCRexServerNameAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x005.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 510.

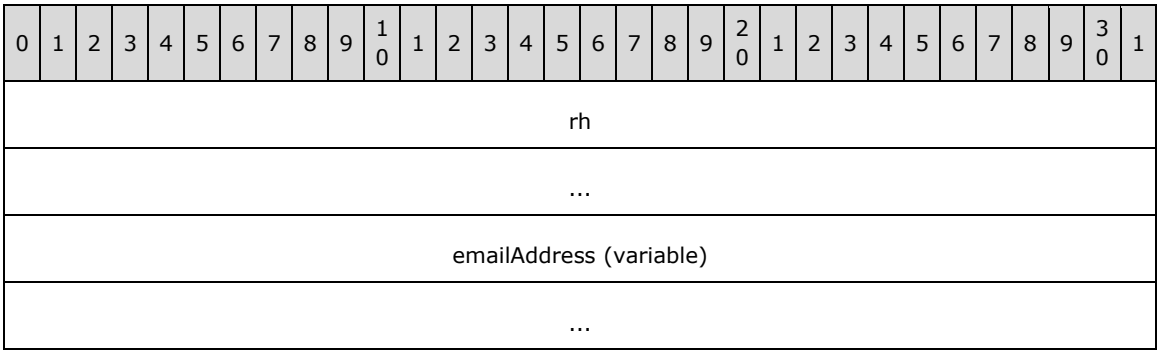
rexServerName (variable): A MachineName that specifies the name of the remote computer to which a camera or microphone is connected to record the video or audio of the corresponding presentation broadcast. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.7 BCEmailAddressAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the e-mail address for audience feedback during a presentation broadcast.

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCEmailAddressAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x006.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 510.

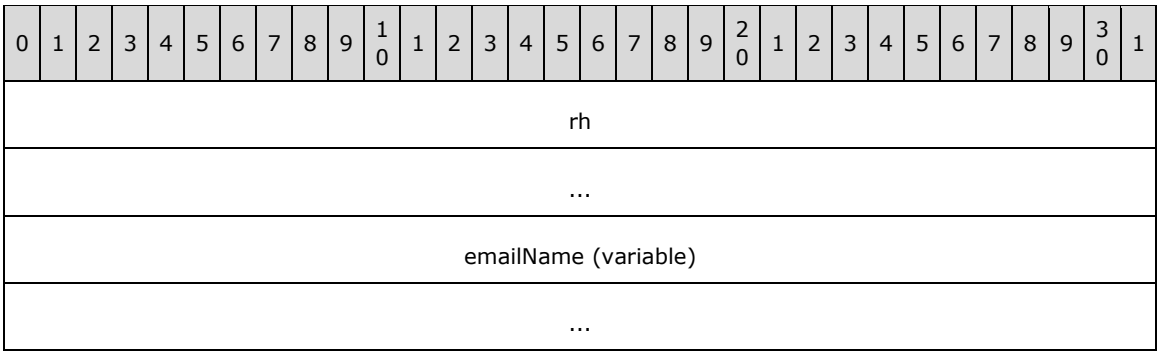
emailAddress (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the e-mail address for audience feedback for the *corresponding presentation broadcast*. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.8 BCEmailNameAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the email name for audience feedback for a presentation broadcast.

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCEmailNameAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x007.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 510.

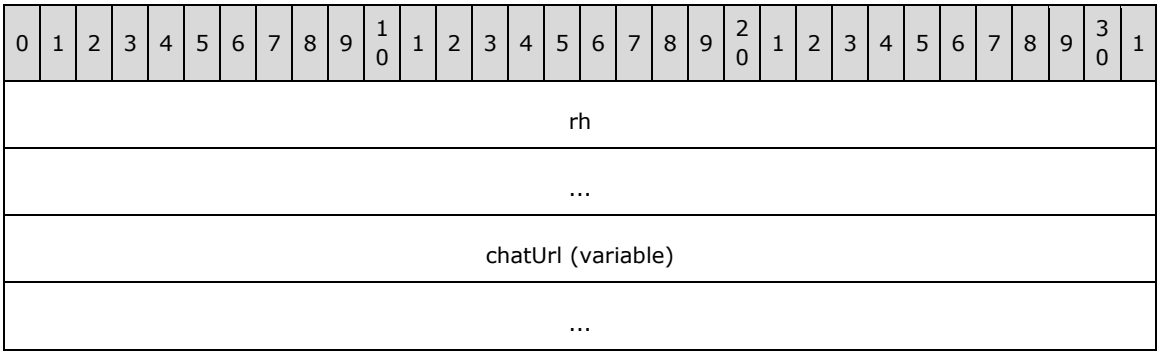
emailName (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the e-mail name for audience feedback for the *corresponding presentation broadcast*. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.9 BCChatUrlAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the URL of a chat server for a presentation broadcast.

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCChatUrlAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x008.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than 4166.

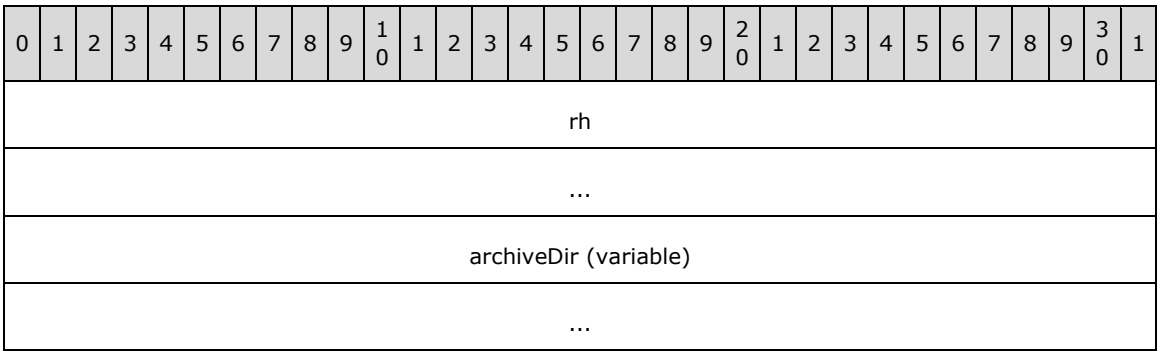
chatUrl (variable): An HttpUrl that specifies the URL of a chat server for the corresponding presentation broadcast. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.10 BCArchiveDirAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the directory location for archival storage of the presentation broadcast.

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCArchiveDirAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x009.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than 510.

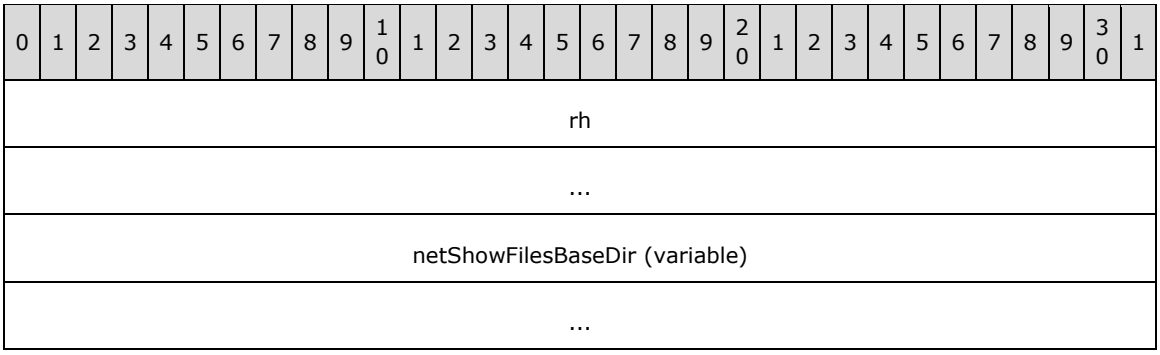
archiveDir (variable): A UncOrLocalPath that specifies the UNC directory location for archival storage of the *corresponding presentation broadcast*. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.11 BCNetShowFilesBaseDirAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the UNC base directory to store presentation broadcast files for NetShow. For more information about NetShow, see [MSFT-UMWNSNS].

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCNetShowFilesBaseDirAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table:

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x00A.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than 510.

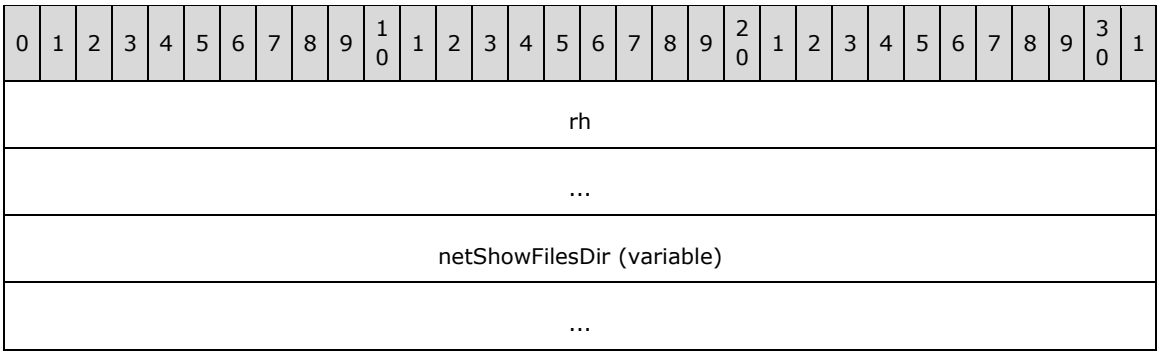
netShowFilesBaseDir (variable): A UncPath that specifies the UNC base directory to store files for NetShow for the *corresponding presentation broadcast*. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.12 BCNetShowFilesDirAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the UNC directory to store presentation broadcast files for NetShow. For more information about NetShow, see [MSFT-UMWNSNS].

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCNetShowFilesDirAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x00B.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than 494.

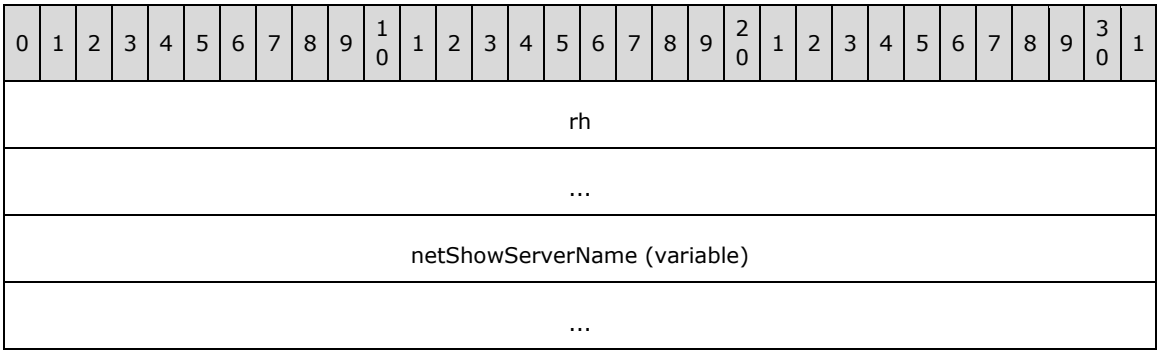
netShowFilesDir (variable): A UncPath that specifies the UNC directory to store files for NetShow for the *corresponding presentation broadcast*. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.13 BCNetShowServerNameAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the name of the NetShow server to use for the presentation broadcast. For more information about NetShow, see [MSFT-UMWNSNS].

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCNetShowServerNameAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x00C.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 510.

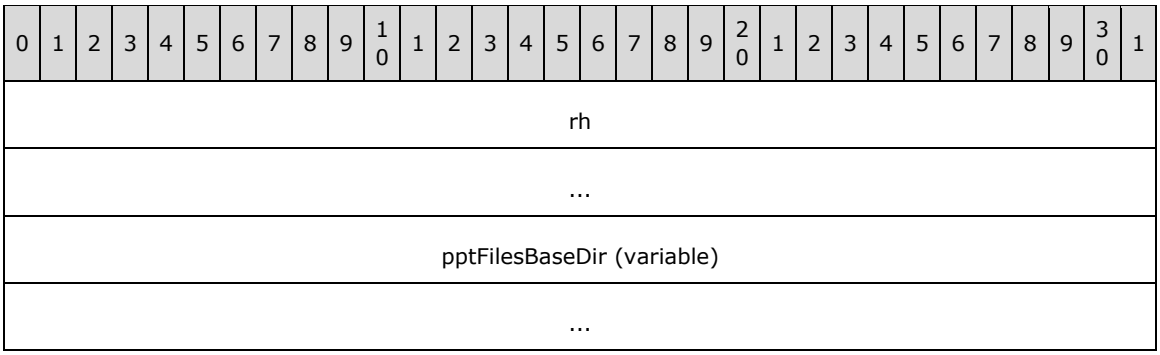
netShowServerName (variable): A MachineName that specifies the name of the NetShow server for the *corresponding presentation broadcast*. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.14 BCPptFilesBaseDirAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the path to the UNC base directory to store presentation broadcast files.

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCPptFilesBaseDirAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x00D.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than 510.

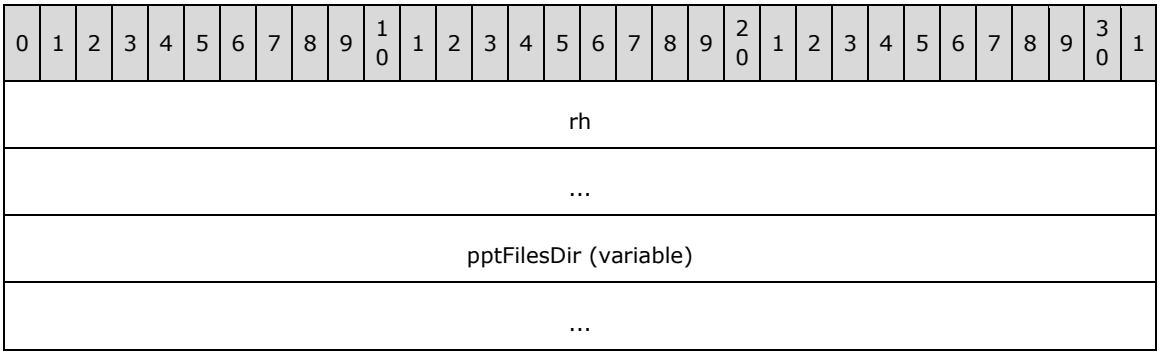
pptFilesBaseDir (variable): A UncPath that specifies the path to the UNC base directory to store files for the *corresponding presentation broadcast*. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.15 BCPptFilesDirAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the path to the UNC directory to store presentation broadcast files.

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCPptFilesDirAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x00E.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than 494.

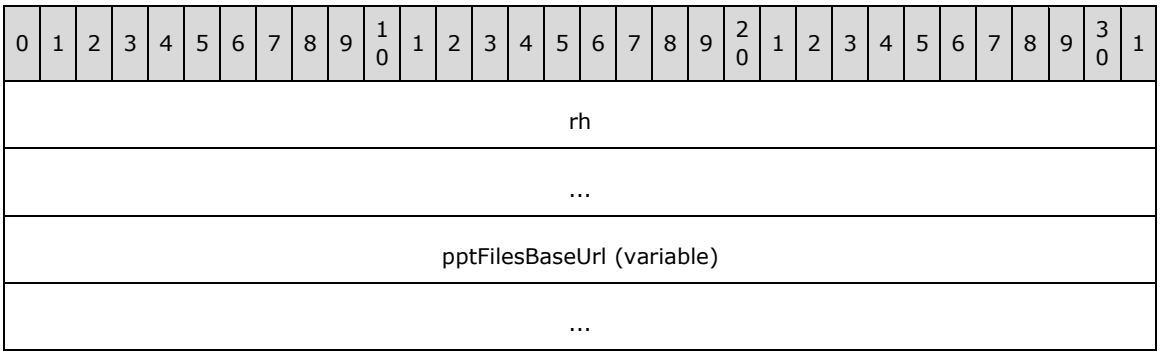
pptFilesDir (variable): A UncPath that specifies the path to the UNC directory to store files for the *corresponding presentation broadcast*. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.16 BCPptFilesBaseUrlAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the location of the presentation broadcast files.

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCPptFilesBaseUrlAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x00F.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than 4118.

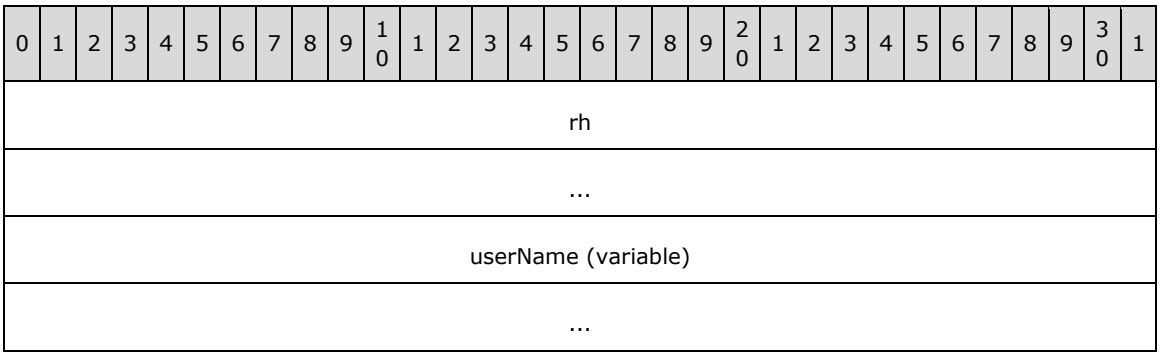
pptFilesBaseUrl (variable): A UncPathOrHttpUrl that specifies the location of the files for the *corresponding presentation broadcast*. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.17 BCUserNameAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the name of the user who scheduled the presentation broadcast.

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCUserNameAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.

Field	Meaning
rh.recInstance	MUST be 0x010.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than 510.

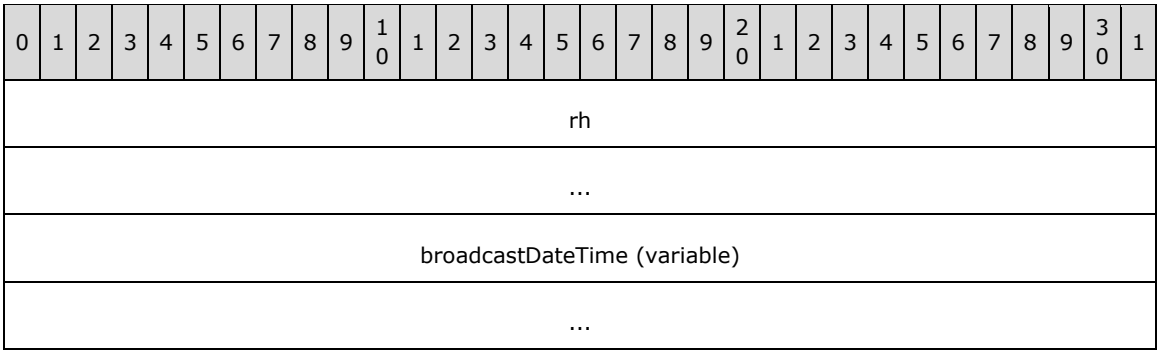
userName (variable): A FileOrDirNameFragment that specifies the name of the user who scheduled the *corresponding presentation broadcast*. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.18 BCBroadcastDateTimeAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the directory name to create under the base directory of a presentation broadcast.

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCBroadcastDateTimeAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x011.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than 520.

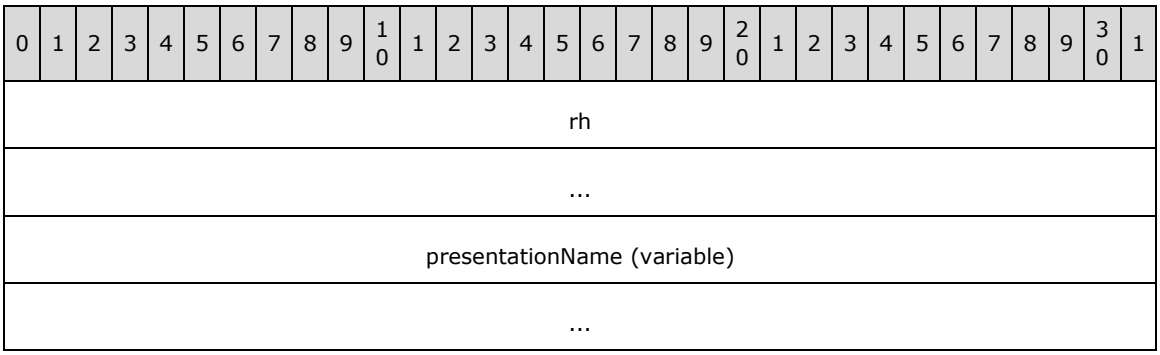
broadcastDateTime (variable): A FileOrDirNameFragment that specifies the directory name to create under the base directory specified in the **bcPptFilesBaseDirAtom** field of the *corresponding presentation broadcast*. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.19 BCPresentationNameAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the name of the presentation in a presentation broadcast.

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCPresentationNameAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x012.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than 510.

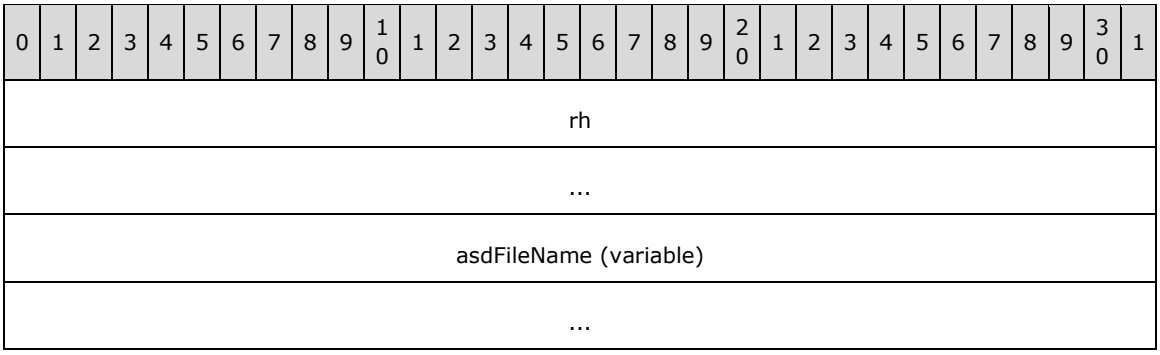
presentationName (variable): A FileOrDirNameFragment that specifies the name of the *corresponding presentation broadcast*. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.20 BCAsdFileNameAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the location of an ASD file for a presentation broadcast. The ASD file is the description file for an Advanced Systems Format (ASF) file, described in [ASF], used to stream audio and video content.

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCAsdFileNameAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x013.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than 520.

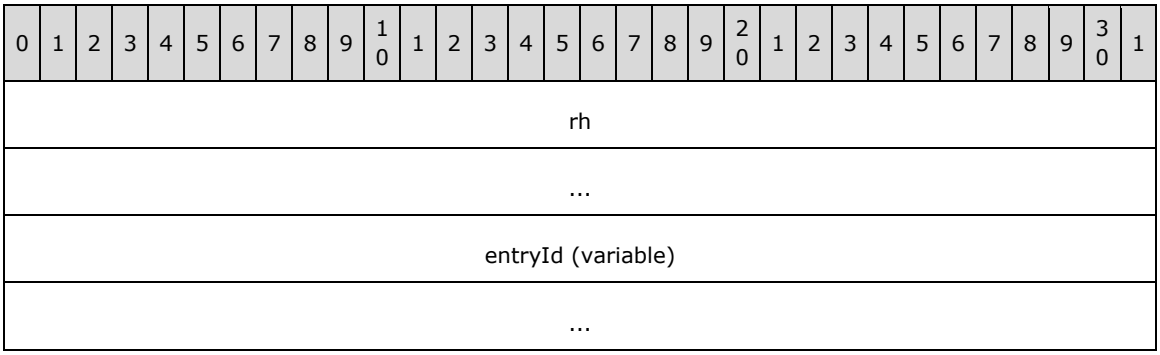
asdFileName (variable): A UncPath that specifies the location of an ASD file for the corresponding presentation broadcast. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.21 BCEntryIDAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies the identifier for a calendar item to associate with this presentation broadcast.

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BCEntryIDAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x014.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number.

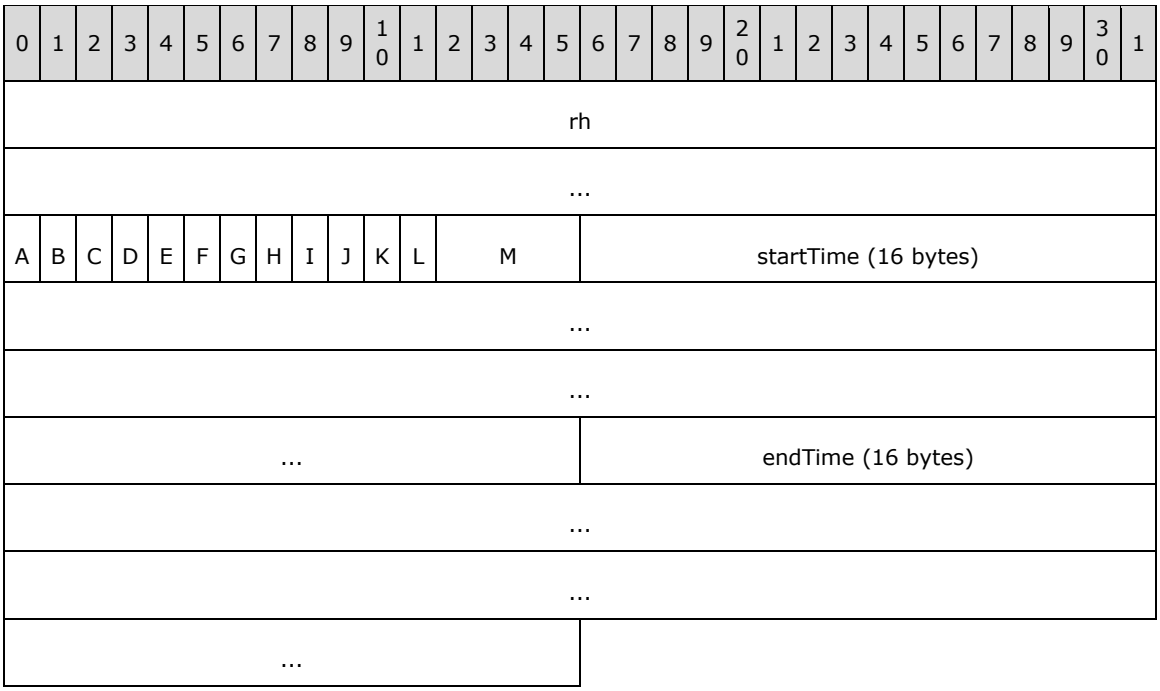
entryId (variable): A UnicodeString that specifies the calendar item identifier for the *corresponding presentation broadcast*. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.17.22 BroadcastDocInfoAtom

Referenced by: BroadcastDocInfo9Container

An atom record that specifies properties of a presentation broadcast.

Let the *corresponding presentation broadcast* be specified by the BroadcastDocInfo2.4.17.1Container record that contains this **BroadcastDocInfoAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_BroadcastDocInfo9Atom.
rh.recLen	MUST be 0x00000022.

A - fSendAudio (1 bit): A bit that specifies whether to include an audio stream.

B - fSendVideo (1 bit): A bit that specifies whether to include a video stream.

C - fCameraRemote (1 bit): A bit that specifies whether the camera is located on a computer other than the computer giving the *corresponding presentation broadcast*.

D - fUseNetShow (1 bit): A bit that specifies whether to use NetShow server technology described in [MSFT-UMWNSNS].

E - fUseOtherServer (1 bit): A bit that specifies whether to use a third-party server for the *corresponding presentation broadcast*.

F - fCanEmail (1 bit): A bit that specifies whether an e-mail address is provided to the audience.

G - fCanChat (1 bit): A bit that specifies whether a chat URL is provided to the audience.

H - fDoArchive (1 bit): A bit that specifies whether the *corresponding presentation broadcast* is archived.

I - fSpeakerNotes (1 bit): A bit that specifies whether the audience can see the speaker notes.

J - fQuarterScreen (1 bit): A bit that specifies whether the slide show is displayed to the presenter in a resizable window.

K - fShowTools (1 bit): A **bit** that specifies whether to show speaker notes to the presenter.

L - fRecordOnly (1 bit): A bit that specifies whether the *corresponding presentation broadcast* is for recording only.

M - reserved (4 bits): MUST be zero and MUST be ignored.

startTime (16 bytes): A DateTimeStruct structure that specifies the time the corresponding presentation broadcast is scheduled to begin.

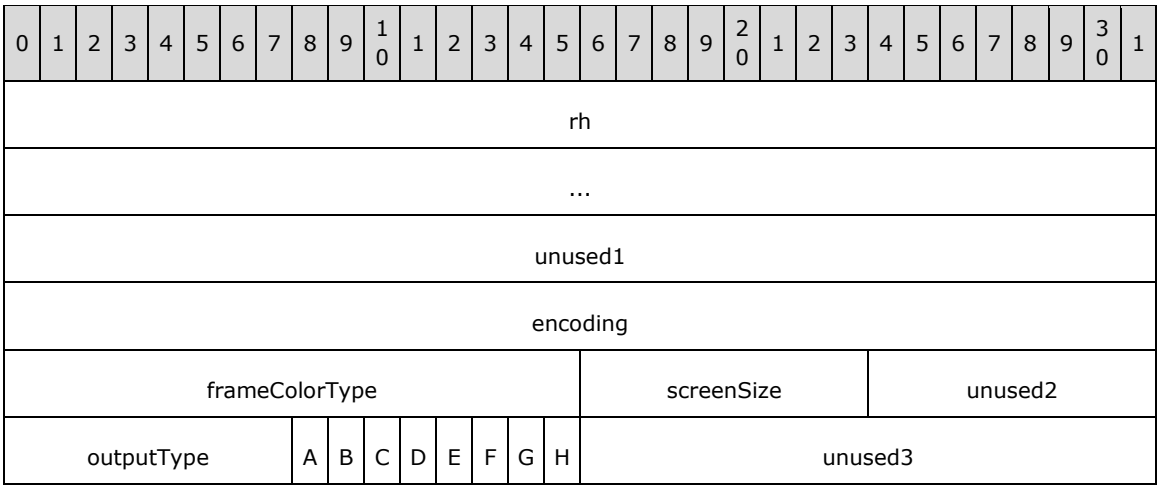
endTime (16 bytes): A DateTimeStruct structure that specifies the time the corresponding presentation broadcast is scheduled to end.

2.4.18 HTML Publish Types

2.4.18.1 HTMLDocInfo9Atom

Referenced by: PP9DocBinaryTagExtension

An atom record that specifies settings for how to publish a document as a Web page.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_HTMLDocInfo9Atom.
rh.recLen	MUST be 0x00000010.

unused1 (4 bytes): Undefined and MUST be ignored.

encoding (4 bytes): An unsigned integer that specifies the **code page** for character encoding used by the Web page. See [MSDN-CP] for a list of possible code pages.

frameColorType (2 bytes): A **WebFrameColorsEnum** enumeration that specifies color options for displaying the text and background for the Web page notes pane and outline pane.

screenSize (1 byte): A **WebScreenSizeEnum** as specified in [MS-OSHARED] section 2.2.1.4 that specifies the document window size for the monitor on which the Web page is displayed.

unused2 (1 byte): Undefined and MUST be ignored.

outputType (1 byte): A **WebOutputEnum** enumeration that specifies the Web browser support that this publication should be optimized for.

A - fShowFrame (1 bit): A bit that specifies whether to include the notes pane and outline pane representation in the Web page.

B - fResizeGraphics (1 bit): A bit that specifies whether the graphics in the Web page are resizable.

C - fOrganizeInFolder (1 bit): A bit that specifies whether any additional files created to represent Web page content in a Web browser are stored in a separate folder.

D - fUseLongFileNames (1 bit): A bit that specifies whether a file name longer than eight characters is valid.

E - fRelyOnVML (1 bit): A bit that specifies whether the Web page requires **Vector Markup Language (VML)** to display in a Web browser.

F - fAllowPNG (1 bit): A bit that specifies whether to save pictures supporting the Web page using **Portable Network Graphics (PNG)** format.

G - fShowSlideAnimation (1 bit): A bit that specifies whether the Web page contains object animation and slide transition effect information.

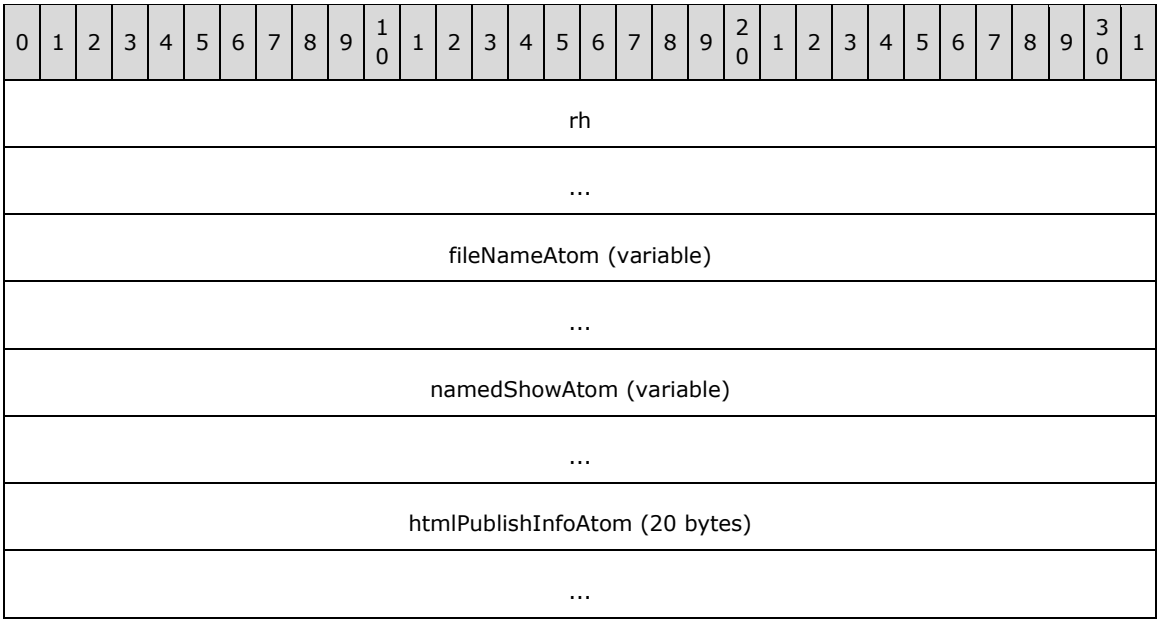
H - reserved (1 bit): MUST be zero and MUST be ignored.

unused3 (2 bytes): Undefined and MUST be ignored.

2.4.18.2 HTMLPublishInfo9Container

Referenced by: PP9DocBinaryTagExtension

A container record that specifies additional information for how to publish a document as a Web page.



...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_HTMLPublishInfo9.

fileNameAtom (variable): A FileNameAtom record that specifies the file name.

namedShowAtom (variable): An optional NamedShowAtom record that specifies the named show being published. It MUST exist if the **htmlPublishInfoAtom.fUseNamedShowX** field is set to **TRUE**.

htmlPublishInfoAtom (20 bytes): A HTMLPublishInfoAtom record that specifies the settings for publishing the document.

2.4.18.3 FileNameAtom

Referenced by: HTMLPublishInfo9Container

An atom record that specifies the file name of the Web page being published.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
fileName (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

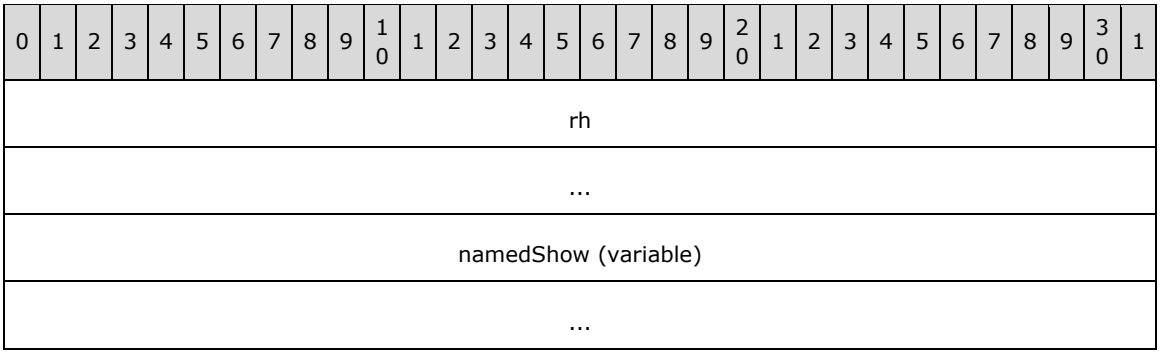
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 510.

fileName (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies a local path, a UNC path, or a URI (specified in [RFC3986]) with the HTTP or FTP scheme. The length, in bytes, of the field is specified by **rh.recLen**. See [MSDN-FILE] for more information about file naming.

2.4.18.4 NamedShowAtom

Referenced by: HTMLPublishInfo9Container

An atom record that specifies the name of a named show that is published to a Web page.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

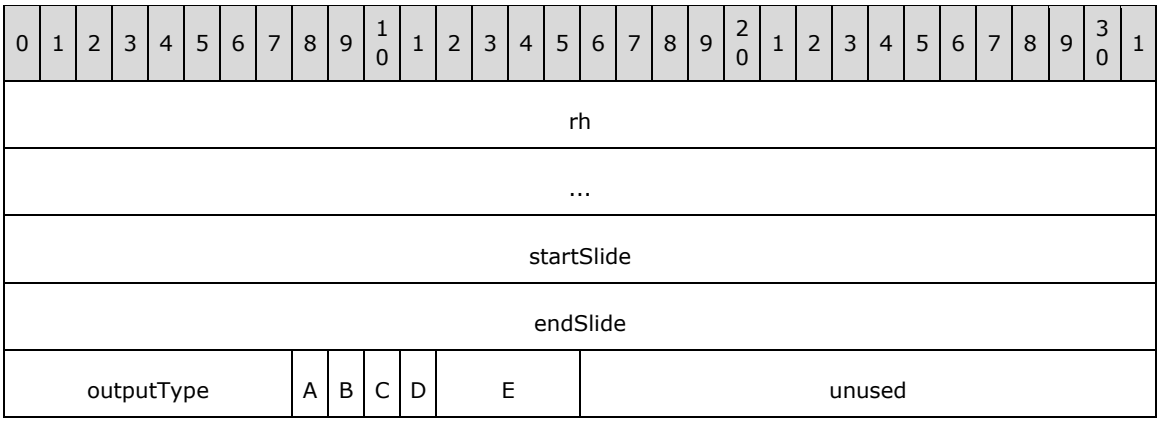
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 62.

namedShow (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the named show to publish. It MUST be the same as the value of the **namedShowName** field of a NamedShowNameAtom record. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.18.5 HTMLPublishInfoAtom

Referenced by: HTMLPublishInfo9Container

An atom record that specifies the settings to publish the document to a Web page.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_HTMLPublishInfoAtom.
rh.recLen	MUST be 0x0000000C.

startSlide (4 bytes): A signed integer that specifies the first slide in the range of slides to publish. It MUST be greater than or equal to zero.

endSlide (4 bytes): A signed integer that specifies the last slide in the range of slides to publish. It MUST be greater than or equal to zero.

outputType (1 byte): A **WebOutputEnum** enumeration that specifies the Web browser support that this publication should be optimized for.

A - fUseSlideRangeX (1 bit): A bit that specifies whether to publish the range of slides defined by **startSlide** and **endSlide**.

B - fUseNamedShowX (1 bit): A bit that specifies whether to publish the slides defined by the **namedShowAtom** field of the HTMLPublishInfo2.4.18.2Container record that contains this **HTMLPublishInfoAtom** record.

C - fLoadInBrowserX (1 bit): A bit that specifies whether to automatically display the Web page in the Web browser.

D - fShowSpeakerNote (1 bit): A bit that specifies whether to display the notes pane when viewing the Web page in a Web browser.

E - reserved (4 bits): MUST be zero and MUST be ignored.

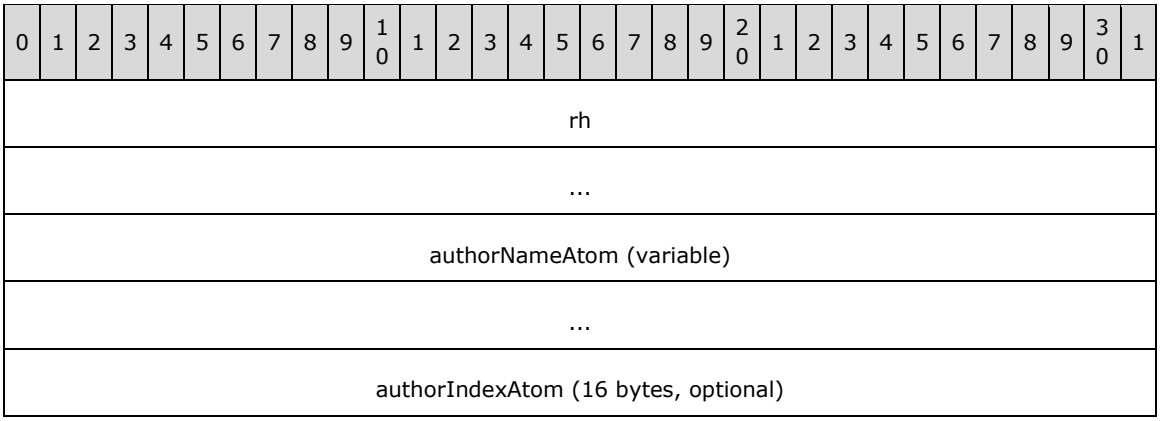
unused (2 bytes): Undefined and MUST be ignored.

2.4.19 Comment Author Types

2.4.19.1 CommentIndex10Container

Referenced by: PP10DocBinaryTagExtension

A container record that specifies information for an author who creates a presentation comment.



...
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_CommentIndex10.

authorNameAtom (variable): An optional AuthorNameAtom record that specifies the name of the author.

authorIndexAtom (16 bytes): An optional CommentIndex2.4.19.3Atom record that specifies an index for deriving a color for the author's presentation comments and an index for the last presentation comment created by the author.

2.4.19.2 AuthorNameAtom

Referenced by: CommentIndex10Container

An atom record that specifies the name of the author who created a presentation comment.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
authorName (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

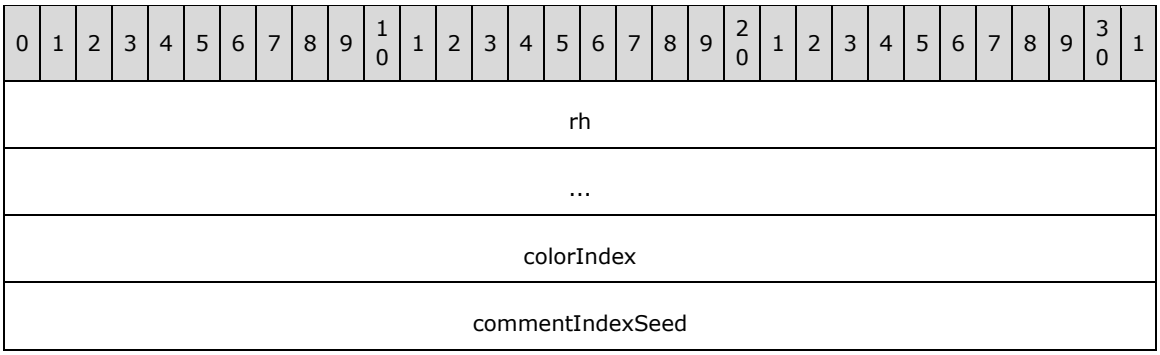
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_CString**.
rh.recLen	MUST be an even number. It MUST be less than or equal to 104.

authorName (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the name of the author. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.19.3 CommentIndex10Atom

Referenced by: CommentIndex10Container

An atom record that specifies an index for deriving a color used to display the author's presentation comments and an index for the last presentation comment created by the author.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_CommentIndex10Atom.
rh.recLen	MUST be 0x00000008.

colorIndex (4 bytes): A signed integer that specifies a zero-based index into the list of colors defined by the rendering application used for displaying the presentation comments created by the author. It MUST be greater than or equal to 0x00000000.

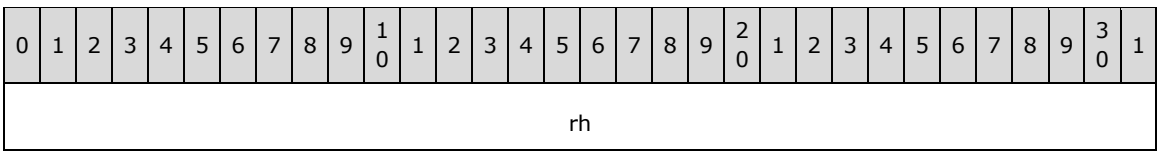
commentIndexSeed (4 bytes): A signed integer that specifies a seed for creating a new index for a presentation comment created by the author. It MUST be greater than or equal to 0x00000000 and MUST be greater than or equal to the value of the **commentAtom.index** field of all Comment2.5.25Container records, where the author name specified by the **commentAuthorAtom** field of the Comment2.5.25Container record matches the author name specified by the **authorNameAtom** field of the CommentIndex2.4.19.1Container record that contains this **CommentIndex10Atom** record.

2.4.20 Document Comparison Types

2.4.20.1 DocToolbarStates10Atom

Referenced by: PP10DocBinaryTagExtension

An atom record that specifies the display options for a reviewing toolbar that has controls that manage presentation comments and the information contained by the DiffTree2.4.20.5Container records and a reviewing gallery that displays the information contained by the DiffTree2.4.20.5Container records.



...			
A	B	reserved	

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_DocToolbarStates10Atom.
rh.recLen	MUST be 0x00000001.

A - fShowReviewingToolbar (1 bit): A bit that specifies whether to display the reviewing toolbar.

B - fShowReviewingGallery (1 bit): A bit that specifies whether to display the reviewing gallery.

reserved (6 bits): MUST be zero and MUST be ignored.

2.4.20.2 SlideListTable10Container

Referenced by: PP10DocBinaryTagExtension

A container record that specifies information about presentation slides contained in the document that also contains this **SlideListTable10Container** record.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
slideListTableSizeAtom																															
...																															
...																															
rgSlideListEntryAtom (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_SlideListTable10.

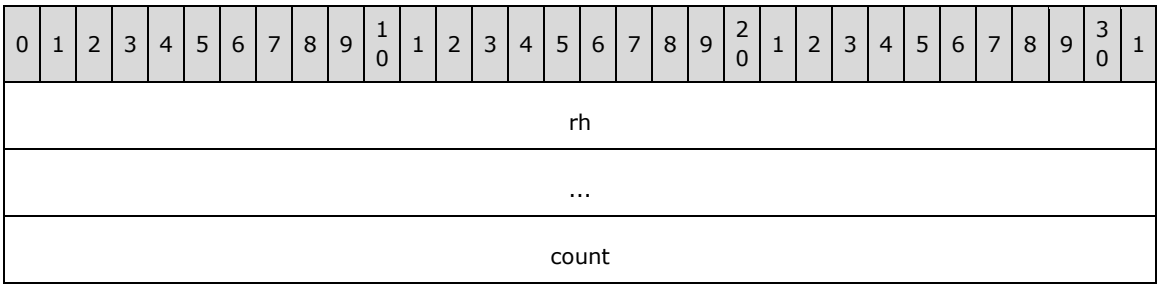
slideListTableSizeAtom (12 bytes): A SlideListTableSize2.4.20.3Atom record that specifies the number of SlideListEntry2.4.20.4Atom records in the **rgSlideListEntryAtom** field.

rgSlideListEntryAtom (variable): An array of SlideListEntry2.4.20.4Atom records that specifies the creation time of the presentation slides in the document. The count of items in the array is specified by the **slideListTableSizeAtom.count** field.

2.4.20.3 SlideListTableSize10Atom

Referenced by: SlideListTable10Container

An atom record that specifies the count of the SlideListEntry2.4.20.4Atom records that are contained within the SlideListTable2.4.20.2Container record that contains this **SlideListTableSize10Atom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

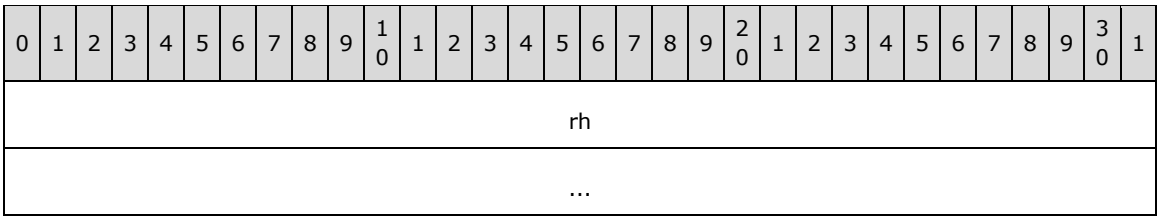
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_SlideListTableSize10Atom.
rh.recLen	MUST be 0x00000004.

count (4 bytes): A signed integer that specifies the count of the **rgSlideListEntryAtom** field of the SlideListTable2.4.20.2Container record that contains this **SlideListTableSize10Atom** record. It MUST be greater than or equal to 0x00000000 and MUST be less than or equal to 0x000F4240.

2.4.20.4 SlideListEntry10Atom

Referenced by: SlideListTable10Container

An atom record that specifies the creation time of a presentation slide in the document that contains this **SlideListEntry10Atom** record.



slideIdRef
dwHighDateTime
dwLowDateTime

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_SlideListEntry10Atom.
rh.recLen	MUST be 0x0000000C.

slideIdRef (4 bytes): A **SlideIdRef** (section 2.2.25) that specifies the presentation slide. Its creation time is specified by the **dwHighDateTime** and **dwLowDateTime** fields of this SlideListEntry10Atom record.

dwHighDateTime (4 bytes): An unsigned integer that specifies the high-order part of the file time, as specified in [MS-DTYP] section 2.3.3.

dwLowDateTime (4 bytes): An unsigned integer that specifies the low-order part of the file time, as specified in [MS-DTYP] section 2.3.3.

2.4.20.5 DiffTree10Container

Referenced by: PP10DocBinaryTagExtension

A container record that specifies the name of a reviewer and how to display the changes to the document made by that reviewer.

Let the *corresponding main master slide* be specified by the **MainMasterContainer** record (section 2.5.3) that is specified by the first **MasterPersistAtom** record (section 2.4.14.2) in the **MasterListWithTextContainer** record (section 2.4.14.1).

Let the *corresponding shape* be specified by the **OfficeArtSpContainer** record ([MS-ODRAW] section 2.2.14) such that the **wzName_complex** property ([MS-ODRAW] section 2.3.4.2) matches the string "Reviewer". The *corresponding shape* is contained by the **drawing** field of the *corresponding main master slide*.

Let the *corresponding OLE object* be specified by the **ExOleEmbedContainer** record (section 2.10.27) whose **exOleObjAtom.exObjId** field matches the **exObjIdRef** field of the ExObjRefAtom record that is contained by the *corresponding shape*.

Let the *corresponding reviewer document* be specified by the *corresponding OLE object*.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															

reviewerNameAtom (variable)
...
docDiff (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_DiffTree10.

reviewerNameAtom (variable): A ReviewerNameAtom record that specifies the name of the reviewer who made the changes to the *corresponding reviewer document*.

docDiff (variable): A DocDiff2.4.20.8Container record that specifies how to display the changes made by the reviewer to the *corresponding reviewer document*.

2.4.20.6 ReviewerNameAtom

Referenced by: DiffTree10Container

An atom record that specifies the name of the reviewer who made changes to a copy of the document that was later merged into this document.

The name of the reviewer MUST be the value of the **GKPIDSI_AUTHOR** property ([MS-OSHARED] section 2.3.3.2.1.1) of the **Summary Info Stream** (section 2.1.4) specified in the *corresponding reviewer document*.

Let the *corresponding reviewer document* be as specified in the DiffTree2.4.20.5Container record that contains this **ReviewerNameAtom** record.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
reviewerName (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.

Field	Meaning
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 104.

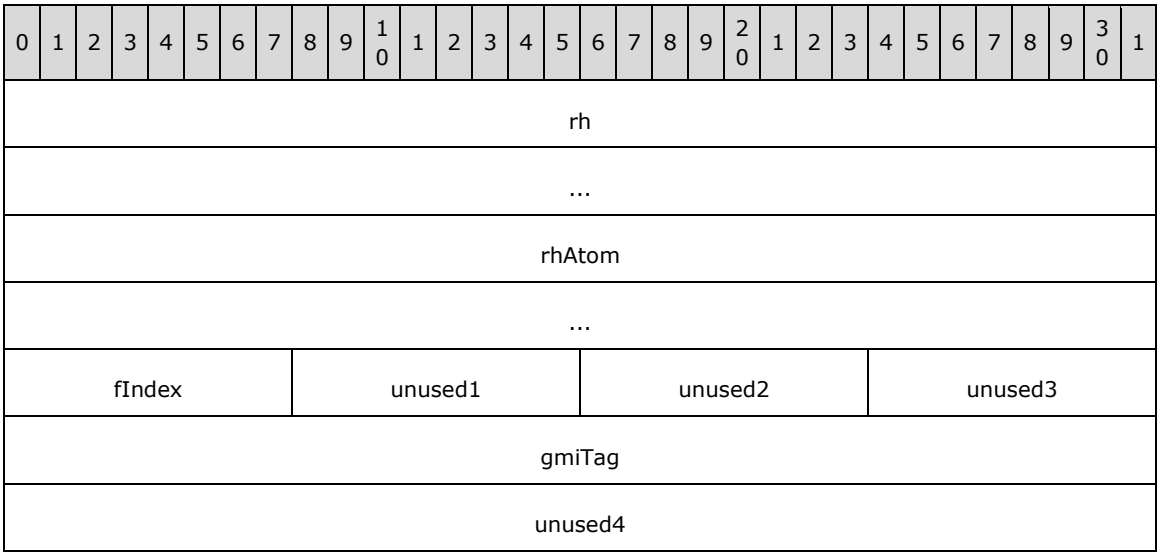
reviewerName (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the name of the reviewer. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.20.7 DiffRecordHeaders

Referenced by: DocDiff10Container, ExternalObjectDiffContainer, HeaderFooterDiffContainer, InteractiveInfoDiffContainer, MainMasterDiffContainer, MasterListDiffContainer, NamedShowDiffContainer, NamedShowListDiffContainer, NotesDiffContainer, RecolorInfoDiffContainer, ShapeDiffContainer, ShapeListDiffContainer, SlideDiffContainer, SlideListDiffContainer, SlideShowDiffContainer, TableDiffContainer, TableListDiffContainer, TextDiffContainer

A structure at the beginning of each container record, when that container record is used to specify how to display the changes to a document made by a reviewer.

Let the *corresponding reviewer document* be as specified in the DiffTree2.4.20.5Container record that contains this **DiffRecordHeaders** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the container record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_Diff10.

rhAtom (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies a header for the atom record that specifies how the changes made by the reviewer are displayed. Sub-fields are further specified in the following table.

Field	Meaning
rhAtom.recVer	MUST be 0x0.
rhAtom.recInstance	MUST be 0x000.
rhAtom.recType	MUST be RT_Diff10Atom.
rhAtom.recLen	MUST be 0x0000000C.

fIndex (1 byte): A **bool1** (section 2.2.2) that specifies instance data. Interpretation of the value is dependent on **gmiTag**.

unused1 (1 byte): Undefined and MUST be ignored.

unused2 (1 byte): Undefined and MUST be ignored.

unused3 (1 byte): Undefined and MUST be ignored.

gmiTag (4 bytes): A **DiffTypeEnum** enumeration that identifies the type of changes made by the reviewer.

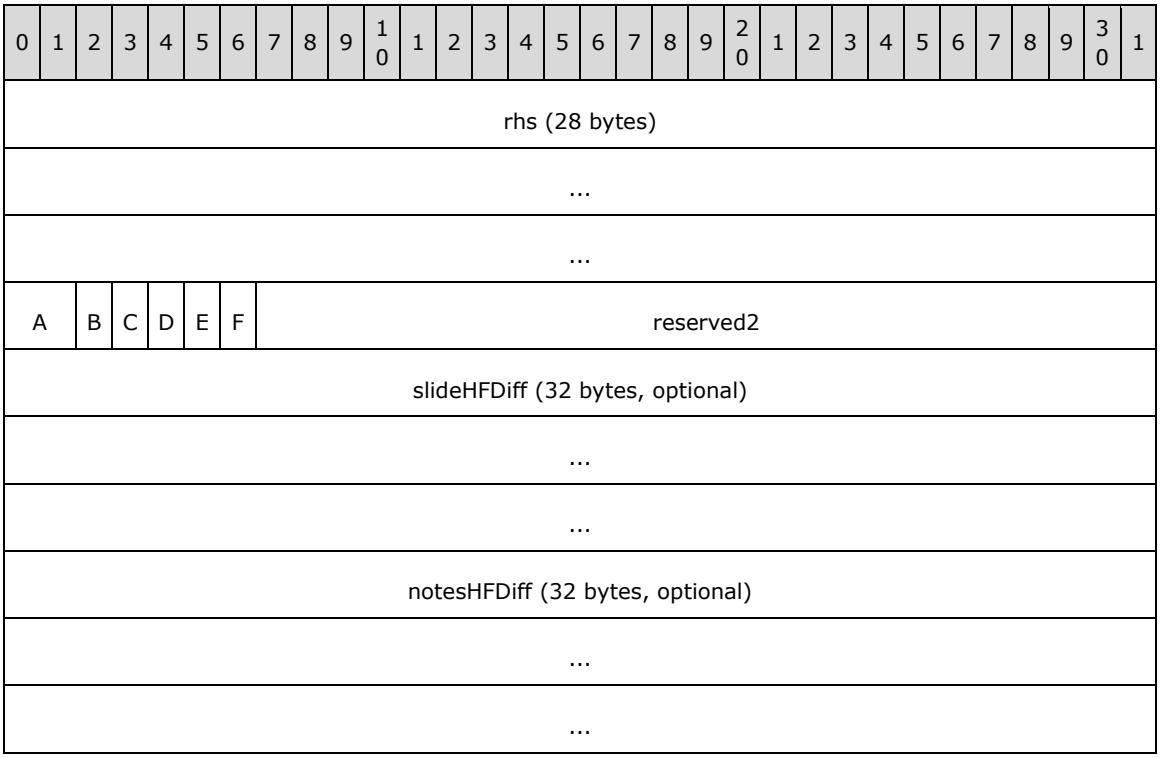
unused4 (4 bytes): Undefined and MUST be ignored.

2.4.20.8 DocDiff10Container

Referenced by: DiffTree10Container

A container record that specifies how to display document-level changes made by the reviewer.

Let the *corresponding reviewer document* be as specified in the DiffTree2.4.20.5Container record that contains this **DocDiff10Container** record.



namedShowListDiff (variable)
...
masterListDiff (variable)
...
slideListDiff (variable)
...

rhs (28 bytes): A DiffRecordHeaders structure that specifies the header for the container record. Sub-fields are further specified in the following table.

Field	Meaning
rhs.fIndex	MUST be 0x00.
rhs.gmiTag	MUST be Diff_DocDiff.

A - reserved1 (2 bits): MUST be zero and MUST be ignored.

B - slideSize (1 bit): A bit that specifies whether the change made by the reviewer in the *corresponding reviewer document* to the **slideSize** field of the **DocumentAtom** record (section 2.4.2) is not displayed.

C - omitTitlePlace (1 bit): A bit that specifies whether the change made by the reviewer in the *corresponding reviewer document* to the **fOmitTitlePlace** field of the **DocumentAtom** record is not displayed.

D - namedShowList (1 bit): A bit that specifies whether the changes made by the reviewer in the *corresponding reviewer document* to any NamedShowContainer record in the **NamedShowsContainer** record (section 2.6.2) are not displayed.

E - slideHeaderFooter (1 bit): A bit that specifies whether the changes made by the reviewer in the *corresponding reviewer document* to the **SlideHeadersFootersContainer** record (section 2.4.15.1) are not displayed.

F - notesHeaderFooter (1 bit): A bit that specifies whether the changes made by the reviewer in the *corresponding reviewer document* to the **NotesHeadersFootersContainer** record (section 2.4.15.6) are not displayed.

reserved2 (25 bits): MUST be zero and MUST be ignored.

slideHFDiff (32 bytes): An optional HeaderFooterDiffContainer record that specifies how to display the changes made by the reviewer in the *corresponding reviewer document* to the **SlideHeadersFootersContainer** record.

notesHFDiff (32 bytes): An optional HeaderFooterDiffContainer record that specifies how to display the changes made by the reviewer in the *corresponding reviewer document* to the **NotesHeadersFootersContainer** record.

namedShowListDiff (variable): An optional NamedShowListDiffContainer record that specifies how to display the changes made by the reviewer in the *corresponding reviewer document* to each NamedShowContainer record in the **NamedShowsContainer** record.

masterListDiff (variable): An optional MasterListDiffContainer record that specifies how to display the changes made by the reviewer in the *corresponding reviewer document* to each **MasterPersistAtom** record (section 2.4.14.2) in the **MasterListWithTextContainer** record (section 2.4.14.1).

slideListDiff (variable): An optional SlideListDiffContainer record that contains records that specify how to display the changes made by the reviewer in the *corresponding reviewer document* to each **SlidePersistAtom** record (section 2.4.14.5) in the **SlideListWithTextContainer** record (section 2.4.14.3).

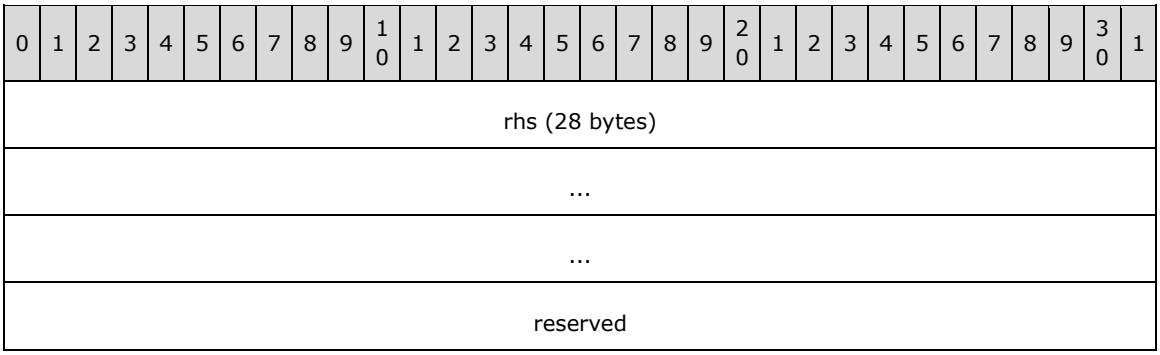
2.4.20.9 HeaderFooterDiffContainer

Referenced by: DocDiff10Container, SlideDiffContainer

A container record that specifies how to display the changes made by the reviewer to the **SlideHeadersFootersContainer** (section 2.4.15.1), **NotesHeadersFootersContainer** (section 2.4.15.6), or PerSlideHeadersFootersContainer record.

Let the *corresponding reviewer document* be as specified in the DiffTree2.4.20.5Container record that contains this **HeaderFooterDiffContainer** record.

Let the *corresponding slide* be as specified in the SlideDiffContainer record that contains this **HeaderFooterDiffContainer** record.



rhs (28 bytes): A DiffRecordHeaders structure that specifies the header for the container record. Sub-fields are further specified in the following table.

Field	Meaning
rhs.fIndex	The value 0x00 specifies display information for the changes made in the *corresponding reviewer document* to the **NotesHeadersFootersContainer** record (section 2.4.15.6). The value 0x01 specifies display information for the changes made in the *corresponding reviewer document* to the **SlideHeadersFootersContainer** record or to the PerSlideHeadersFootersContainer record in the *corresponding slide*.
rhs.gmiTag	MUST be Diff_HeaderFooterDiff.

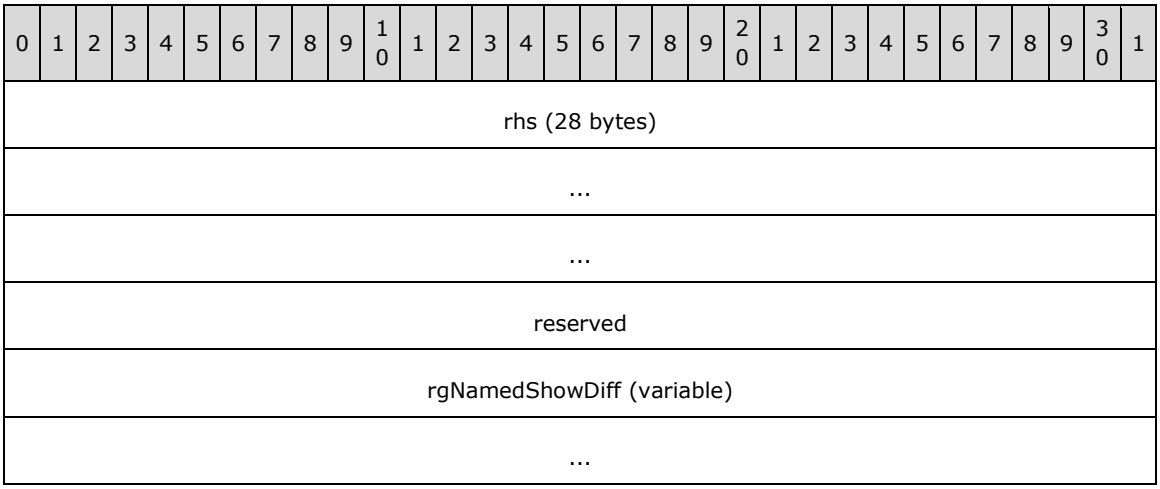
reserved (32 bits): MUST be zero and MUST be ignored.

2.4.20.10 NamedShowListDiffContainer

Referenced by: DocDiff10Container

A container record that specifies how to display the changes made by the reviewer to the **NamedShowsContainer** record (section 2.6.2).

Let the *corresponding reviewer document* be as specified in the DiffTree2.4.20.5Container record that contains this **NamedShowListDiffContainer** record.



rhs (28 bytes): A DiffRecordHeaders structure that specifies the header for the container record. Sub-fields are further specified in the following table.

Field	Meaning
rhs.fIndex	MUST be 0x00.
rhs.gmiTag	MUST be Diff_NamedShowListDiff.

reserved (32 bits): MUST be zero and MUST be ignored.

rgNamedShowDiff (variable): An array of NamedShowDiffContainer records that specifies how to display changes made by the reviewer in the *corresponding reviewer document* to the named shows. The size, in bytes, of the array is specified by the following formula:

```
rhs.rh.recLen - rhs.rhAtom.recLen - 8
```

2.4.20.11 NamedShowDiffContainer

Referenced by: NamedShowListDiffContainer

A container record that specifies how to display the changes made by the reviewer to a NamedShowContainer record.

Let the *corresponding reviewer document* be as specified in the DiffTree2.4.20.5Container record that contains this **NamedShowDiffContainer** record.

The i^{th} **NamedShowDiffContainer** record in its parent NamedShowListDiffContainer record specifies how to display changes made to the i^{th} NamedShowContainer record in its parent **NamedShowsContainer** record (section 2.6.2) in the *corresponding reviewer document*.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rhs (28 bytes)																															
...																															
...																															
reserved																															

rhs (28 bytes): A DiffRecordHeaders structure that specifies the header for the container record. Sub-fields are further specified in the following table.

Field	Meaning
rhs.fIndex	MUST be 0x00.
rhs.gmiTag	MUST be Diff_NamedShowDiff.

reserved (32 bits): MUST be zero and MUST be ignored.

2.4.20.12 SlideListDiffContainer

Referenced by: DocDiff10Container

A container record that specifies how to display the changes made by the reviewer to the **SlideListWithTextContainer** record (section 2.4.14.3).

Let the *corresponding reviewer document* be as specified in the DiffTree2.4.20.5Container record that contains this **SlideListDiffContainer** record.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rhs (28 bytes)																															
...																															
...																															
reserved																															
rgSlideDiff (variable)																															
...																															

rhs (28 bytes): A DiffRecordHeaders structure that specifies the header for the container record. Sub-fields are further specified in the following table.

Field	Meaning
rhs.fIndex	MUST be 0x00.
rhs.gmiTag	MUST be Diff_SlideListDiff.

reserved (32 bits): MUST be zero and MUST be ignored.

rgSlideDiff (variable): An array of SlideDiffContainer records that specifies how to display changes made by the reviewer in the *corresponding reviewer document* to the **SlideListWithTextContainer** record (section 2.4.14.3). The size, in bytes, of the array is specified by the following formula:

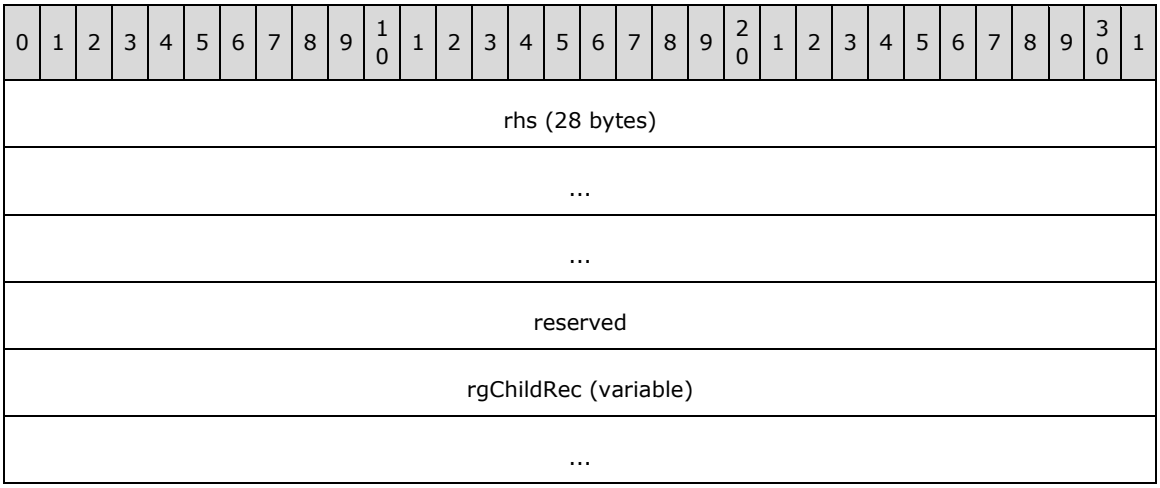
```
rhs.rh.recLen - rhs.rhAtom.recLen - 8
```

2.4.20.13 MasterListDiffContainer

Referenced by: DocDiff10Container

A container record that specifies how to display the changes made by the reviewer to the **MasterListWithTextContainer** record (section 2.4.14.1).

Let the *corresponding reviewer document* be as specified in the DiffTree2.4.20.5Container record that contains this **MasterListDiffContainer** record.



rhs (28 bytes): A DiffRecordHeaders structure that specifies the header for the container record. Sub-fields are further specified in the following table.

Field	Meaning
rhs.fIndex	MUST be 0x00.
rhs.gmiTag	MUST be Diff_MasterListDiff.

reserved (32 bits): MUST be zero and MUST be ignored.

rgChildRec (variable): An array of MasterListDiff2.4.20.14ChildContainer records that specifies how to display changes made by the reviewer in the *corresponding reviewer document* to the **MasterListWithTextContainer** record (section 2.4.14.1). The size, in bytes, of the array is specified by the following formula:

```
rhs.rh.recLen - rhs.rhAtom.recLen - 8
```

2.4.20.14 MasterListDiff10ChildContainer

Referenced by: MasterListDiffContainer

A variable type record whose type and meaning are dictated by the value of **rhs.gmiTag** as specified in the following table.

Value	Meaning
Diff_SlideDiff	A SlideDiffContainer record that specifies how to display the changes to a title master slide.
Diff_MainMasterDiff	A MainMasterDiffContainer record that specifies how to display the changes to a main master slide.

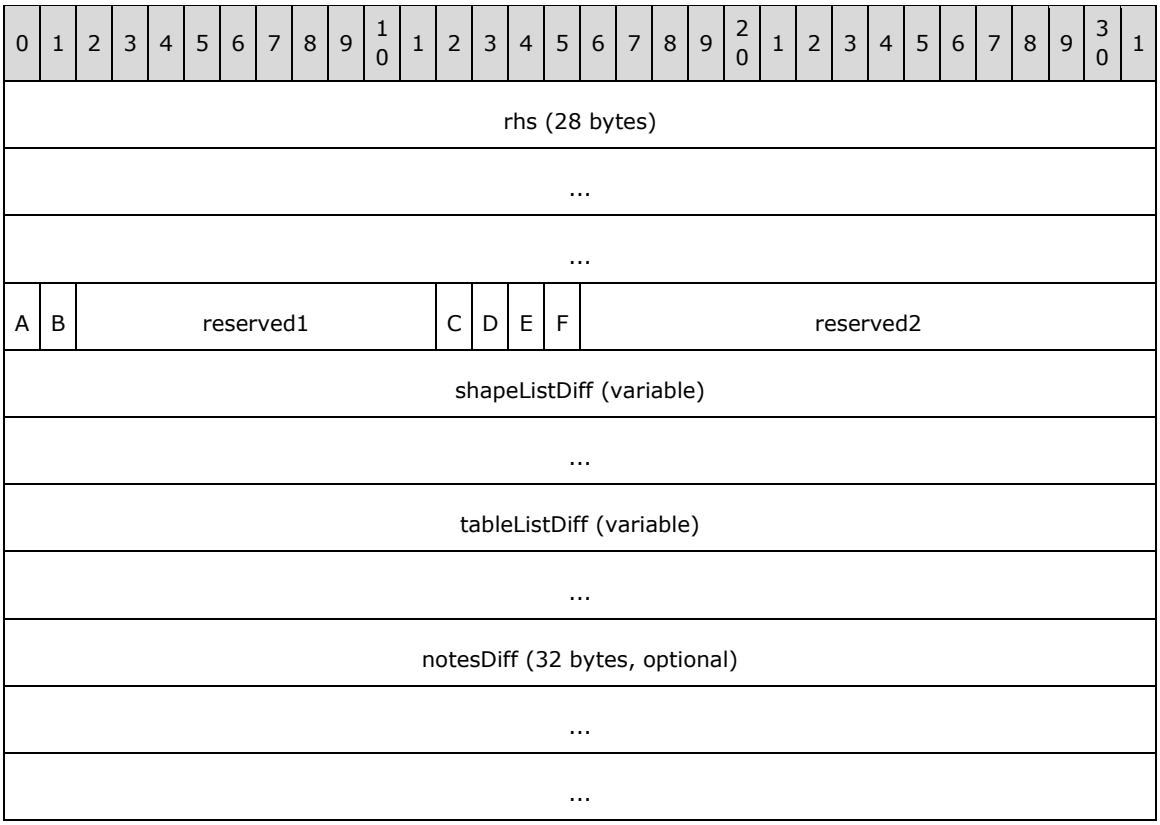
2.4.20.15 MainMasterDiffContainer

Referenced by: MasterListDiff10ChildContainer

A container record that specifies how to display the changes made by the reviewer to a main master slide.

Let the *corresponding reviewer document* be as specified in the DiffTree2.4.20.5Container record that contains this **MainMasterDiffContainer** record.

The i[th] **MainMasterDiffContainer** record in its parent MasterListDiffContainer record specifies how to display changes made to the **MainMasterContainer** record in the *corresponding reviewer document* that is referenced by the i[th] **MasterPersistAtom** record (section 2.4.14.2) in its parent **MasterListWithTextContainer** record (section 2.4.14.1) in the *corresponding reviewer document*. Let the *corresponding main master slide* be the **MainMasterContainer** record so specified.



rhs (28 bytes): A DiffRecordHeaders structure that specifies the header for the container record. Sub-fields are further specified in the following table.

Field	Meaning
rhs.fIndex	MUST be 0x00.
rhs.gmiTag	MUST be Diff_MainMasterDiff.

A - scheme (1 bit): A **bit** that specifies whether the change made by the reviewer to the **slideFlags.fMasterScheme** field of the SlideAtom record contained within the corresponding main master slide is not displayed.

B - background (1 bit): A **bit** that specifies whether the change made by the reviewer to the **slideFlags.fMasterBackground** field of the SlideAtom record contained within the corresponding main master slide is not displayed.

reserved1 (10 bits): MUST be zero and MUST be ignored.

C - timeNode (1 bit): A **bit** that specifies whether the change made by the reviewer to the **ExtTimeNodeContainer** record (section 2.8.15) contained within the *corresponding main master slide* is not displayed.

D - addMainMaster (1 bit): A **bit** that specifies whether the addition of the *corresponding main master slide* made by the reviewer in the *corresponding reviewer document* is not displayed.

E - deleteMainMaster (1 bit): A **bit** that specifies whether the deletion of the *corresponding main master slide* made by the reviewer in the *corresponding reviewer document* is not displayed.

F - locked (1 bit): A **bit** that specifies whether the change made by the reviewer to the **slideFlagsAtom.fPreserveMaster** field of the PP2.5.24SlideBinaryTagExtension record contained within the *corresponding main master slide* is not displayed.

reserved2 (16 bits): MUST be zero and MUST be ignored.

shapeListDiff (variable): An optional ShapeListDiffContainer record that specifies how to display the changes made by the reviewer to the shapes contained within the *corresponding main master* slide.

tableListDiff (variable): An optional TableListDiffContainer record that specifies how to display the changes made by the reviewer to the **table objects** contained within the *corresponding main master slide*.

notesDiff (32 bytes): An optional NotesDiffContainer record that specifies how to display the changes made by the reviewer in the *corresponding reviewer document* to the notes master slide.

2.4.20.16 SlideDiffContainer

Referenced by: MasterListDiff10ChildContainer, SlideListDiffContainer

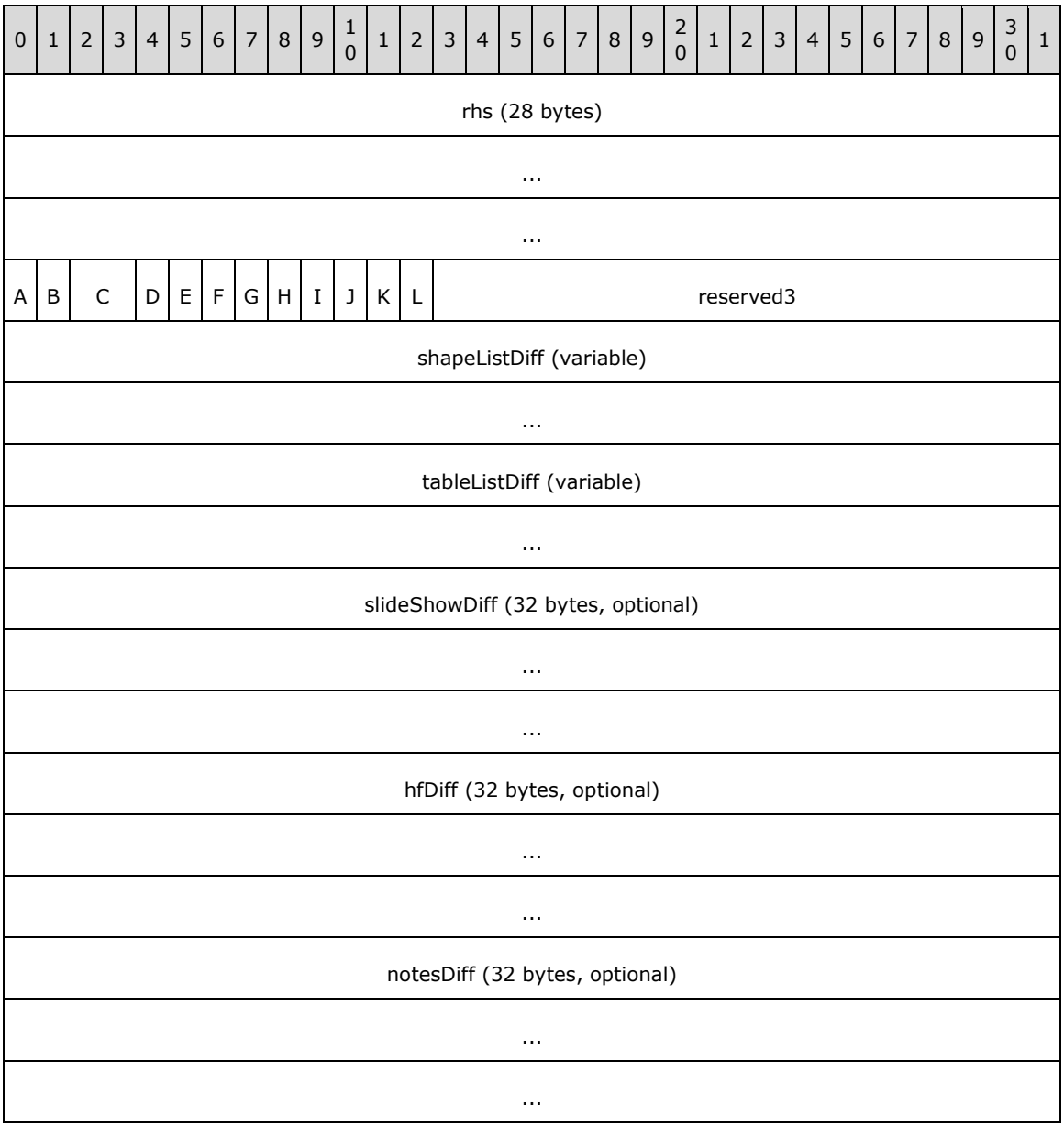
A container record that specifies how to display the changes made by the reviewer to a title master slide or to a presentation slide.

Let the *corresponding reviewer document* be as specified in the DiffTree2.4.20.5Container record that contains this **SlideDiffContainer** record.

When this **SlideDiffContainer** record is contained within a MasterListDiffContainer record, the i^{th} **SlideDiffContainer** record in its parent MasterListDiffContainer record specifies how to display changes made to the **SlideContainer** record (section 2.5.1) in the *corresponding reviewer document* that is referenced by the i^{th} **MasterPersistAtom** record (section 2.4.14.2) in its parent

MasterListWithTextContainer record (section 2.4.14.1) in the *corresponding reviewer document*. Let the *corresponding slide* be the **SlideContainer** record (section 2.5.1) so specified.

When this **SlideDiffContainer** record is contained within a SlideListDiffContainer, the i^{th} **SlideDiffContainer** record in its parent SlideListDiffContainer record specifies how to display changes made to the **SlideContainer** record in the *corresponding reviewer document* that is specified by the i^{th} **SlidePersistAtom** record (section 2.4.14.5) in its parent **SlideListWithTextContainer** record (section 2.4.14.3) in the *corresponding reviewer document*. Let the *corresponding slide* be the **SlideContainer** record so specified.



rhs (28 bytes): A DiffRecordHeaders structure that specifies the header for the container record. Sub-fields are further specified in the following table.

Field	Meaning
rhs.fIndex	MUST be 0x00.

Field	Meaning
rhs.gmiTag	MUST be Diff_SlideDiff.

A - scheme (1 bit): A bit that specifies whether the change made by the reviewer to the **slideFlags.fMasterScheme** field of the SlideAtom record contained within the *corresponding slide* is not displayed.

B - background (1 bit): A bit that specifies whether the change made by the reviewer to the **slideFlags.fMasterBackground** field of the SlideAtom record contained within the *corresponding slide* is not displayed.

C - reserved1 (2 bits): MUST be zero and MUST be ignored.

D - addSlide (1 bit): A bit that specifies whether the addition of the *corresponding slide* made by the reviewer in the *corresponding reviewer document* is not displayed.

E - deleteSlide (1 bit): A bit that specifies whether the deletion of the *corresponding slide* made by the reviewer in the *corresponding reviewer document* is not displayed.

F - layout (1 bit): A bit that specifies whether the change made by the reviewer to the **geom** field of the SlideAtom record contained within the *corresponding slide* is not displayed.

G - slideShow (1 bit): A bit that specifies whether the changes made by the reviewer to the SlideShowSlideInfoAtom record contained within the *corresponding slide* are not displayed.

H - headerFooter (1 bit): A bit that specifies whether the changes made by the reviewer in the *corresponding reviewer document* to the PerSlideHeadersFootersContainer record in the *corresponding slide* are not displayed.

I - reserved2 (1 bit): MUST be zero and MUST be ignored.

J - master (1 bit): A bit that specifies whether the change made by the reviewer to the **masterIdRef** field of the SlideAtom record contained within the *corresponding slide* is not displayed.

K - position (1 bit): A bit that specifies whether the change made by the reviewer to the position of the *corresponding slide* in the **SlideListWithTextContainer** record (section 2.4.14.3) in the *corresponding reviewer document* is not displayed.

L - timeNode (1 bit): A bit that specifies whether the change made by the reviewer to the **ExtTimeNodeContainer** record (section 2.8.15) contained within the *corresponding slide* is not displayed.

reserved3 (19 bits): MUST be zero and MUST be ignored.

shapeListDiff (variable): An optional ShapeListDiffContainer record that specifies how to display the changes made by the reviewer to the shapes contained within the *corresponding slide*.

tableListDiff (variable): An optional TableListDiffContainer record that specifies how to display the changes made by the reviewer to the table objects contained within the *corresponding slide*.

slideShowDiff (32 bytes): An optional SlideShowDiffContainer record that specifies how to display the changes made by the reviewer to the SlideShowSlideInfoAtom record contained within the *corresponding slide*.

hfDiff (32 bytes): An optional HeaderFooterDiffContainer record that specifies how to display the changes made by the reviewer to the PerSlideHeadersFootersContainer record contained within the *corresponding slide*.

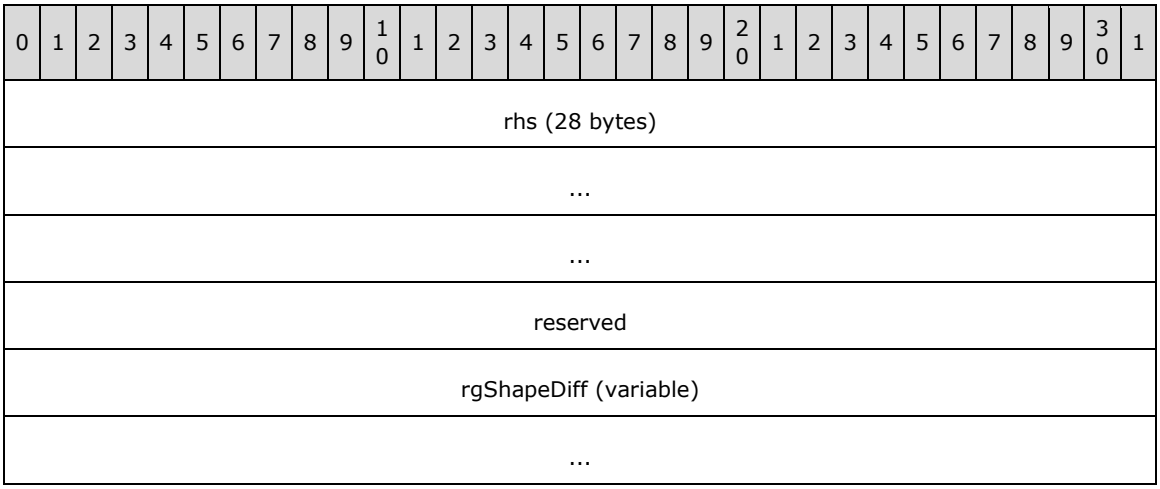
notesDiff (32 bytes): An optional NotesDiffContainer record that specifies how to display the changes made by the reviewer to the notes slide for the *corresponding slide*.

2.4.20.17 ShapeListDiffContainer

Referenced by: MainMasterDiffContainer, SlideDiffContainer

A container record that specifies how to display the changes made by the reviewer to shapes.

Let the *corresponding slide* or *corresponding main master slide* be as specified in the SlideDiffContainer or MainMasterDiffContainer record that contains this **ShapeListDiffContainer** record.



rhs (28 bytes): A DiffRecordHeaders structure that specifies the header for the container record. Sub-fields are further specified in the following table.

Field	Meaning
rhs.fIndex	MUST be 0x00.
rhs.gmiTag	MUST be Diff_ShapeListDiff.

reserved (32 bits): MUST be zero and MUST be ignored.

rgShapeDiff (variable): An array of ShapeDiffContainer records that specifies how to display changes made by the reviewer to the shapes contained within the *corresponding slide* or *corresponding main master slide*. The size, in bytes, of the array is specified by the following formula:

```
rhs.rh.recLen - rhs.rhAtom.recLen - 8
```

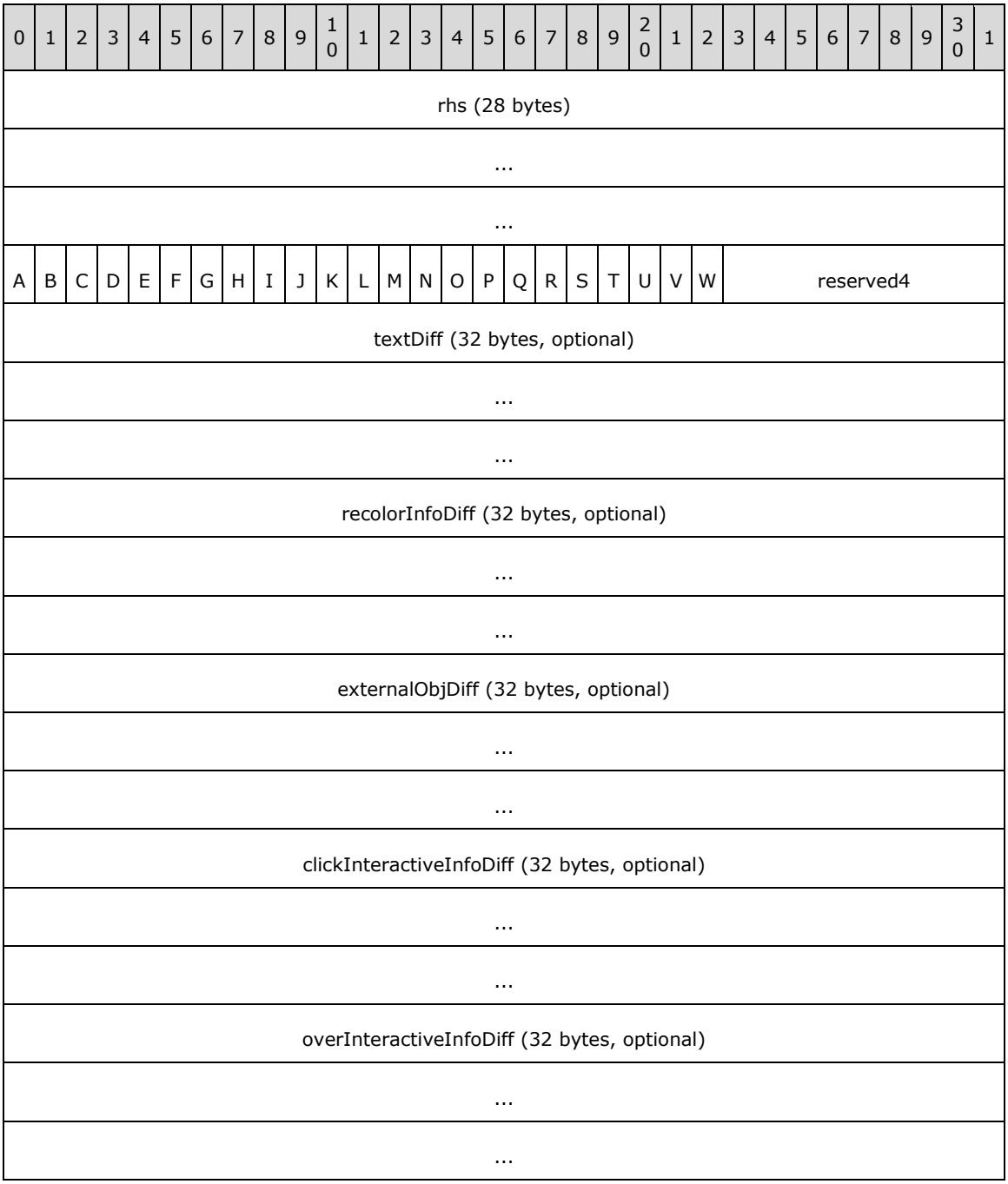
2.4.20.18 ShapeDiffContainer

Referenced by: ShapeListDiffContainer

A container record that specifies how to display the changes made by the reviewer to a shape.

Let the *corresponding slide* or *corresponding main master slide* be as specified in the ShapeListDiffContainer record that contains this **ShapeDiffContainer** record.

The i[th] **ShapeDiffContainer** record in its parent ShapeListDiffContainer specifies how to display changes made to the i[th] **OfficeArtSpContainer** record ([MS-ODRAW] section 2.2.14) contained within the *corresponding slide* or *corresponding main master slide*. Let the **OfficeArtSpContainer** record so specified be the *corresponding shape*.



rhs (28 bytes): A DiffRecordHeaders structure that specifies the header for the container record. Sub-fields are further specified in the following table.

Field	Meaning
rhs.fIndex	MUST be 0x00.

Field	Meaning
rhs.gmiTag	MUST be Diff_ShapeDiff.

A - addShape (1 bit): A **bit** that specifies whether the addition of the *corresponding shape* made by the reviewer in the *corresponding slide* or *corresponding main master slide* is not displayed.

B - deleteShape (1 bit): A **bit** that specifies whether the deletion of the *corresponding shape* made by the reviewer in the *corresponding slide* or *corresponding main master slide* is not displayed.

C - child (1 bit): A **bit** that specifies whether the change made by the reviewer to the set of child shapes of the *corresponding shape* is not displayed.

MUST be ignored if the **shapeProp.fGroup** field of the **OfficeArtSpContainer** record ([MS-ODRAW] section 2.2.14) is **FALSE**.

D - position (1 bit): A bit that specifies whether the change made by the reviewer to the position of the *corresponding shape* in the **OfficeArtDgContainer** record ([MS-ODRAW] section 2.2.13) is not displayed.

E - recolorInfo (1 bit): A bit that specifies whether the changes made by the reviewer to the RecolorInfoAtom record contained within the *corresponding shape* are not displayed.

F - externalObject (1 bit): A bit that specifies whether the changes made by the reviewer to the *corresponding external object* referenced from within the *corresponding shape* is not displayed.

Let the *corresponding external object* be an external object that is specified by either the **ExMediaAtom** record (section 2.10.6) or the **ExOleObjAtom** record (section 2.10.12) whose **exObjId** field equals the **exObjIdRef** field of ExObjRefAtom record contained within the *corresponding shape*.

G - interactiveInfoOnOver (1 bit): A bit that specifies whether the changes made by the reviewer to the MouseOverInteractiveInfoContainer record contained within the *corresponding shape* are not displayed.

H - interactiveInfoOnClick (1 bit): A bit that specifies whether the changes made by the reviewer to the MouseClickInteractiveInfoContainer record contained within the *corresponding shape* are not displayed.

I - reserved1 (1 bit): MUST be zero and MUST be ignored.

J - msopsid3DSettings (1 bit): A bit that specifies whether the changes made by the reviewer to the 3D object ([MS-ODRAW] section 2.3.15), 3D Style ([MS-ODRAW] section 2.3.16), and perspective style ([MS-ODRAW] section 2.3.14) properties of the *corresponding shape* are not displayed.

K - msopsidBWSettings (1 bit): A bit that specifies whether the changes made by the reviewer to the **bWMode** ([MS-ODRAW] section 2.3.2.3), **bWModePureBW** ([MS-ODRAW] section 2.3.2.4), and **bWModeBW** ([MS-ODRAW] section 2.3.2.5) properties of the *corresponding shape* are not displayed.

L - msopsidAutoShape (1 bit): A bit that specifies whether the changes made by the reviewer to the shape type in the **OfficeArtFSP** record ([MS-ODRAW] section 2.2.40) and the callout ([MS-ODRAW] section 2.3.3) properties of the *corresponding shape* are not displayed.

M - msopsidLineStyle (1 bit): A bit that specifies whether the changes made by the reviewer to the line style properties ([MS-ODRAW] section 2.3.8) of the *corresponding shape* are not displayed.

N - msopsidFillStyle (1 bit): A bit that specifies whether the changes made by the reviewer to the fill style properties ([MS-ODRAW] section 2.3.7) of the *corresponding shape* are not displayed.

O - msopsidShadowStyle (1 bit): A bit that specifies whether the changes made by the reviewer to the shadow style properties ([MS-ODRAW] section 2.3.13) of the *corresponding shape* are not displayed.

P - msopsidWordArt (1 bit): A bit that specifies whether the changes made by the reviewer to the geometry text properties ([MS-ODRAW] section 2.3.22) of the *corresponding shape* are not displayed.

Q - msopsidPicture (1 bit): A bit that specifies whether the changes made by the reviewer to the blip properties ([MS-ODRAW] section 2.3.23) of the *corresponding shape* are not displayed.

R - msopsidOrientation (1 bit): A bit that specifies whether the changes made by the reviewer to the transform properties ([MS-ODRAW] section 2.3.18 and [MS-ODRAW] section 2.3.19) of the *corresponding shape* are not displayed.

S - msopsidTextSetting (1 bit): A bit that specifies whether the changes made by the reviewer to the text properties ([MS-ODRAW] section 2.3.21) of the *corresponding shape* are not displayed.

T - reserved2 (1 bit): MUST be zero and MUST be ignored.

U - msopsidSize (1 bit): A bit that specifies whether the changes made by the reviewer to the OfficeArtClientAnchor record contained within the *corresponding shape* are not displayed.

V - reserved3 (1 bit): MUST be zero and MUST be ignored.

W - ruler (1 bit): A bit that specifies whether the changes made by the reviewer to the TextRulerAtom record of the OfficeArtClientTextbox record contained within the corresponding shape are not displayed.

reserved4 (9 bits): MUST be zero and MUST be ignored.

textDiff (32 bytes): An optional TextDiffContainer record that specifies how to display the changes made by the reviewer to the OfficeArtClientTextbox record contained within the corresponding shape.

recolorInfoDiff (32 bytes): An optional RecolorInfoDiffContainer record that specifies how to display the changes made by the reviewer to the RecolorInfoAtom record contained within the *corresponding shape*.

externalObjDiff (32 bytes): An optional ExternalObjectDiffContainer record that specifies how to display the changes made by the reviewer to the external object referenced from within the *corresponding shape*.

clickInteractiveInfoDiff (32 bytes): An optional InteractiveInfoDiffContainer record that specifies how to display the changes made by the reviewer to the MouseClickInteractiveInfoContainer record contained within the *corresponding shape*.

overInteractiveInfoDiff (32 bytes): An optional InteractiveInfoDiffContainer record that specifies how to display the changes made by the reviewer to the MouseOverInteractiveInfoContainer record contained within the *corresponding shape*.

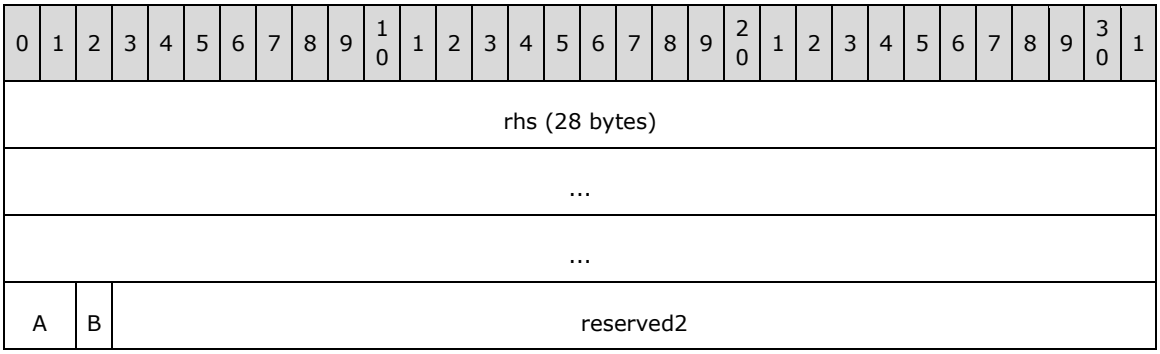
2.4.20.19 TextDiffContainer

Referenced by: ShapeDiffContainer

A container record that specifies how to display the changes made by the reviewer to the text of a shape.

Let the *corresponding shape* be as specified in the ShapeDiffContainer record that contains this **TextDiffContainer** record.

Let the *corresponding text* be as specified in the OfficeArtClientTextbox record contained within the *corresponding shape*.



rhs (28 bytes): A DiffRecordHeaders structure that specifies the header for the container record. Sub-fields are further specified in the following table.

Field	Meaning
rhs.fIndex	MUST be 0x00.
rhs.gmiTag	MUST be Diff_TextDiff.

A - reserved1 (2 bits): MUST be zero and MUST be ignored.

B - wordList (1 bit): A **bit** that specifies whether the changes made by the reviewer to the *corresponding text* are not displayed.

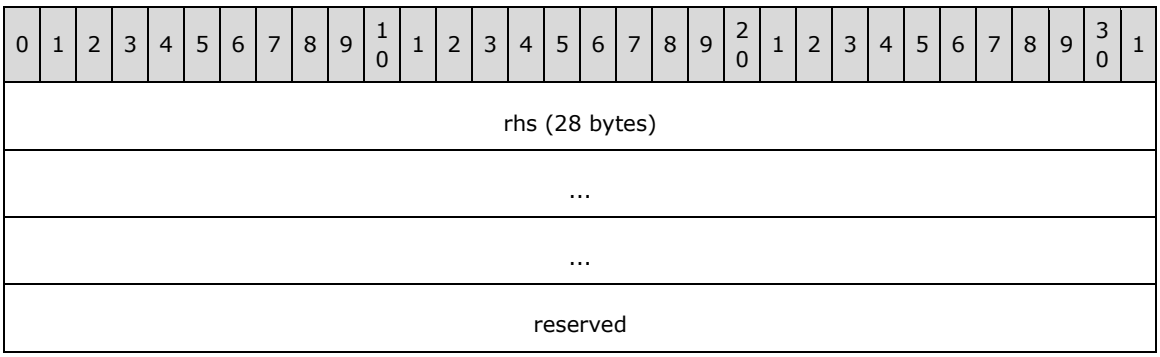
reserved2 (29 bits): MUST be zero and MUST be ignored.

2.4.20.20 RecolorInfoDiffContainer

Referenced by: ShapeDiffContainer

A container record that specifies how to display the changes made by the reviewer to the RecolorInfoAtom record contained within the *corresponding shape*.

Let the *corresponding shape* be as specified in the ShapeDiffContainer record that contains this **RecolorDiffContainer** record.



rhs (28 bytes): A DiffRecordHeaders structure that specifies the header for the container record. Sub-fields are further specified in the following table.

Field	Meaning
rhs.fIndex	MUST be 0x00.
rhs.gmiTag	MUST be Diff_RecolorInfoDiff.

reserved (32 bits): MUST be zero and MUST be ignored.

2.4.20.21 ExternalObjectDiffContainer

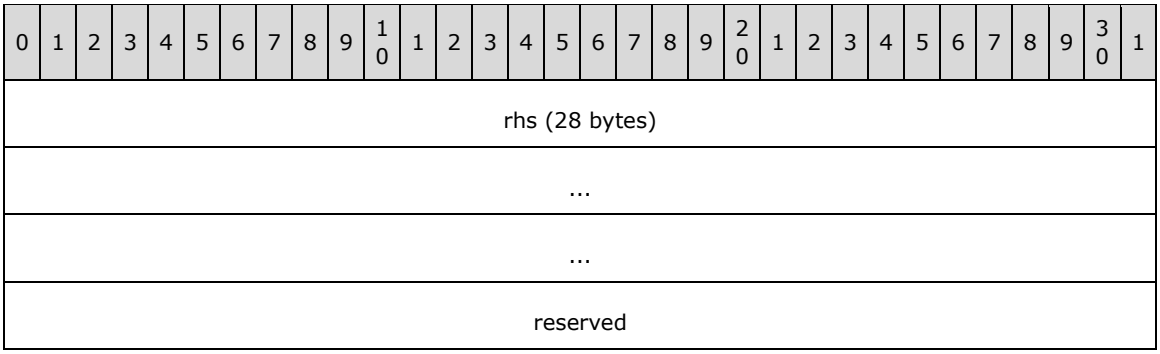
Referenced by: ShapeDiffContainer

A container record that specifies how to display the changes made by the reviewer to an external object.

Let the *corresponding shape* be as specified in the ShapeDiffContainer record that contains this **ExternalObjectDiffContainer** record.

Let the *corresponding external object* be specified by one of the container records listed in the following table that contains a field, also listed in the table, that matches the **exObjIdRef** field of the ExObjRefAtom record contained within the *corresponding shape*.

External object container	Field
ExAviMovieContainer	**exVideo.exMediaAtom.exObjId**
ExMCIMovieContainer	**exVideo.exMediaAtom.exObjId**
ExCDAudioContainer	**exMediaAtom.exObjId**
ExMIDIAudioContainer	**exMediaAtom.exObjId**
ExWAVAudioEmbeddedContainer	**exMedia.exObjId**
ExWAVAudioLinkContainer	**exMedia.exObjId**
ExControlContainer (section 2.10.10)	**exOleObjAtom.exObjId**
ExOleEmbedContainer (section 2.10.27)	**exOleObjAtom.exObjId**
ExOleLinkContainer (section 2.10.29)	**exOleObjAtom.exObjId**



rhs (28 bytes): A DiffRecordHeaders structure that specifies the header for the container record. Sub-fields are further specified in the following table.

Field	Meaning
rhs.fIndex	MUST be 0.

Field	Meaning
rhs.gmiTag	MUST be Diff_ExternalObjectDiff.

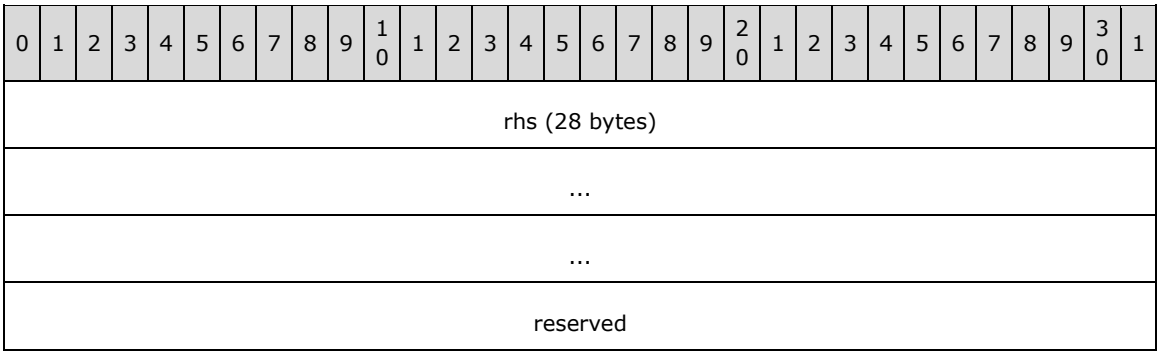
reserved (32 bits): MUST be zero and MUST be ignored.

2.4.20.22 InteractiveInfoDiffContainer

Referenced by: ShapeDiffContainer

A container record that specifies how to display the changes made by the reviewer to the MouseClickInteractiveInfoContainer or MouseOverInteractiveInfoContainer record contained within the *corresponding shape*.

Let the *corresponding shape* be as specified in the ShapeDiffContainer record that contains this **InteractiveInfoDiffContainer** record.



rhs (28 bytes): A DiffRecordHeaders structure that specifies the header for the container record. Sub-fields are further specified in the following table.

Field	Meaning
rhs.fIndex	The value 0x00 specifies display information for the changes to the MouseOverInteractiveInfoContainer record contained within the *corresponding shape*. The value 0x01 specifies display information for the changes to the MouseClickInteractiveInfoContainer record contained within the *corresponding shape*.
rhs.gmiTag	MUST be Diff_InteractiveInfoDiff.

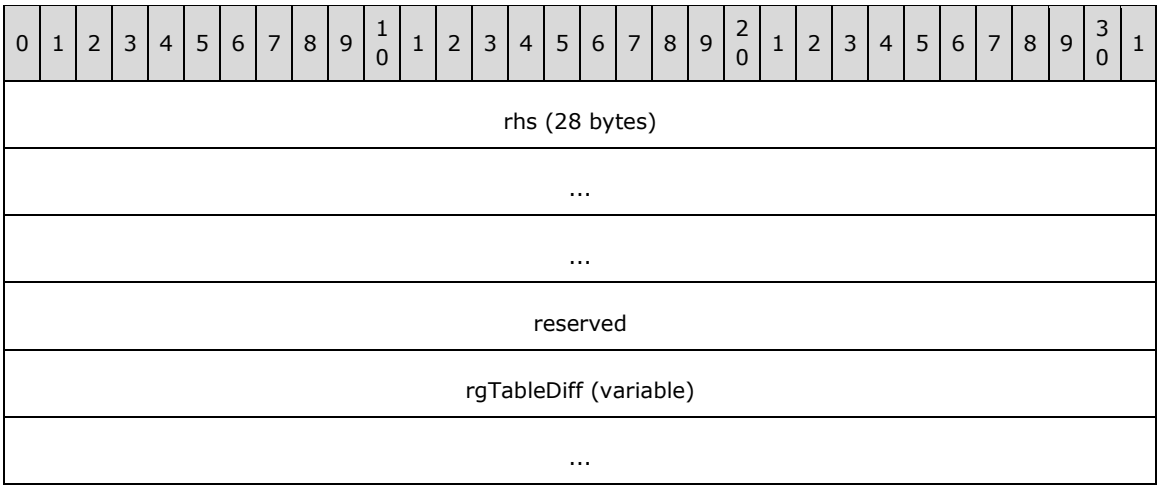
reserved (32 bits): MUST be zero and MUST be ignored.

2.4.20.23 TableListDiffContainer

Referenced by: MainMasterDiffContainer, SlideDiffContainer

A container record that specifies how to display the changes made by the reviewer to table objects.

Let the *corresponding slide* or *corresponding main master slide* be as specified in the SlideDiffContainer or MainMasterDiffContainer record that contains this **TableListDiffContainer** record.



rhs (28 bytes): A DiffRecordHeaders structure that specifies the header for the container record. Sub-fields are further specified in the following table.

Field	Meaning
rhs.fIndex	MUST be 0x00.
rhs.gmiTag	MUST be Diff_TableListDiff.

reserved (32 bits): MUST be zero and MUST be ignored.

rgTableDiff (variable): An array of TableDiffContainer records that specifies how to display changes made by the reviewer to the table objects in the *corresponding slide* or *corresponding main master slide*. The size, in bytes, of the array is specified by the following formula:

```
rhs.rh.recLen -rhs.rhAtom.recLen - 8
```

2.4.20.24 TableDiffContainer

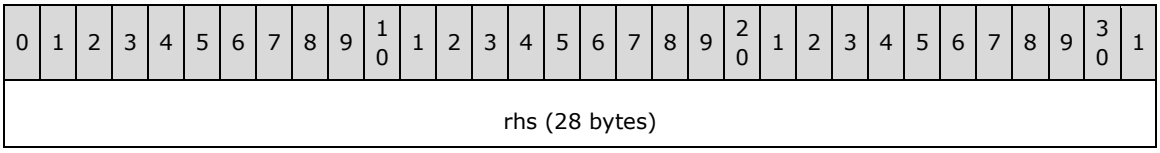
Referenced by: TableListDiffContainer

A container record that specifies how to display the changes made by the reviewer to a table object.

Let the *corresponding slide* or *corresponding main master slide* be as specified in the TableListDiffContainer record that contains this **TableDiffContainer** record.

The i^{th} **TableDiffContainer** record in its parent TableListDiffContainer record specifies how to display changes made to the i^{th} *corresponding table object* contained within the *corresponding slide* or *corresponding main master slide*.

Let the *corresponding table object* be specified by the **OfficeArtSpContainer** record ([MS-ODRAW] section 2.2.14) such that the **tableProperties.fIsTable** field of the **tableProperties** property ([MS-ODRAW] section 2.3.4.36) MUST be **TRUE**.



...				
...				
A	B	C	D	reserved

rhs (28 bytes): A DiffRecordHeaders structure that specifies the header for the container record. Sub-fields are further specified in the following table:

Field	Meaning
rhs.fIndex	MUST be 0x00.
rhs.gmiTag	MUST be Diff_TableDiff.

A - addTable (1 bit): A **bit** that specifies whether the addition of the *corresponding table object* made by the reviewer in the *corresponding slide* or *corresponding main master slide* is not displayed.

B - deleteTable (1 bit): A **bit** that specifies whether the deletion of the *corresponding table object* made by the reviewer in the *corresponding slide* or *corresponding main master slide* is not displayed.

C - modifiedTable (1 bit): A **bit** that specifies whether the changes made by the reviewer to the *corresponding table object* are not displayed.

D - position (1 bit): A **bit** that specifies whether the change made by the reviewer to the z-order of the *corresponding table object* is not displayed.

reserved (28 bits): MUST be zero and MUST be ignored.

2.4.20.25 SlideShowDiffContainer

Referenced by: SlideDiffContainer

A container record that specifies how to display changes made by the reviewer to the SlideShowSlideInfoAtom record contained within the *corresponding slide*.

Let the *corresponding slide* be as specified in the SlideDiffContainer record that contains this **SlideShowDiffContainer** record.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rhs (28 bytes)																															
...																															
...																															
reserved																															

rhs (28 bytes): A DiffRecordHeaders structure that specifies the header for the container record. Sub-fields are further specified in the following table.

Field	Meaning
rhs.fIndex	MUST be 0x00.
rhs.gmiTag	MUST be Diff_SlideShowDiff.

reserved (32 bits): MUST be zero and MUST be ignored.

2.4.20.26 NotesDiffContainer

Referenced by: MainMasterDiffContainer, SlideDiffContainer

A container record that specifies how to display the changes made by the reviewer to the notes master slide or to a notes slide.

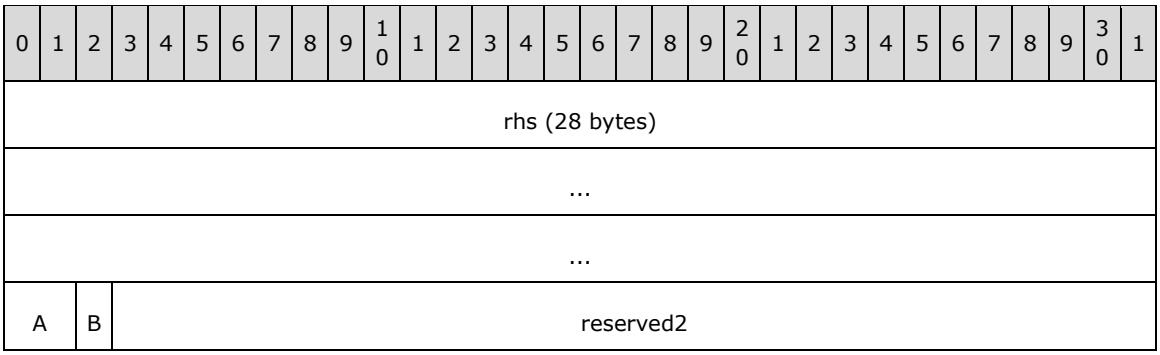
Let the *corresponding reviewer document* be as specified in the DiffTree2.4.20.5Container record that contains this **NotesDiffContainer** record.

Let the *corresponding slide* be as specified in the SlideDiffContainer record that contains this **NotesDiffContainer** record or let the *corresponding main master slide* be as specified in the MainMasterDiffContainer record that contains this **NotesDiffContainer** record.

Let the *corresponding notes slide* be the **NotesContainer** record (section 2.5.6) as specified by the **slideAtom.notesIdRef** field of the *corresponding slide* or let the *corresponding notes master slide* be as specified by the **notesMasterPersistIdRef** field of the **DocumentAtom** record (section 2.4.2) in the corresponding reviewer document.

Let the *corresponding notes shape* be specified by the **OfficeArtSpContainer** record ([MS-ODRAW] section 2.2.14) contained in the *corresponding notes slide* or *corresponding notes master slide* such that the **placementId** field of the PlaceholderAtom record contained within the **OfficeArtSpContainer** record has the value of PT_NotesBody.

Let the *corresponding notes text* be as specified in the OfficeArtClientTextbox record contained within the *corresponding notes shape*.



rhs (28 bytes): A DiffRecordHeaders structure that specifies the header for the container record. Sub-fields are further specified in the following table:

Field	Meaning
rhs.fIndex	MUST be 0x00.
rhs.gmiTag	MUST be Diff_NotesDiff.

A - reserved1 (2 bits): MUST be zero and MUST be ignored.

B - wordList (1 bit): A **bit** that specifies whether the change made by the reviewer to the *corresponding notes text* is not displayed. It MUST be **FALSE** if the *corresponding notes text* is contained within the *corresponding notes master slide*.

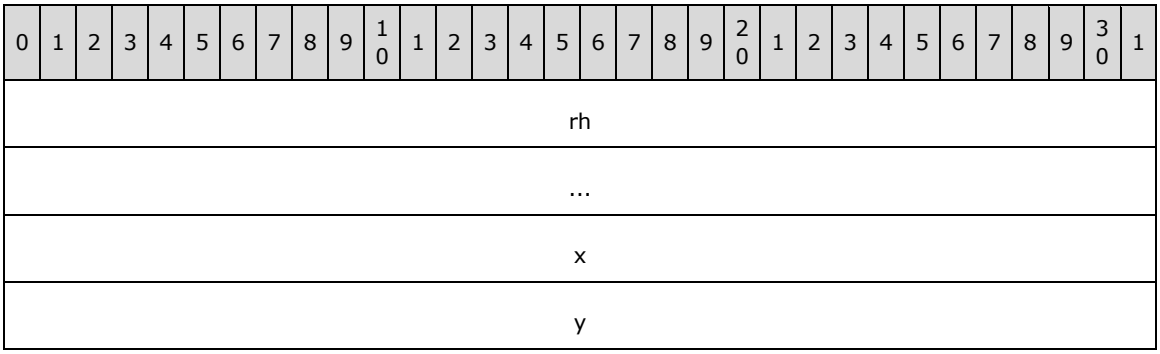
reserved2 (29 bits): MUST be zero and MUST be ignored.

2.4.21 View Info Types

2.4.21.1 GridSpacing10Atom

Referenced by: PP10DocBinaryTagExtension

An atom record that specifies spacing for a grid that can be used to align objects on a slide and to display positioning cues. Only square grids are allowed.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_GridSpacing10Atom.
rh.recLen	MUST be 0x00000008.

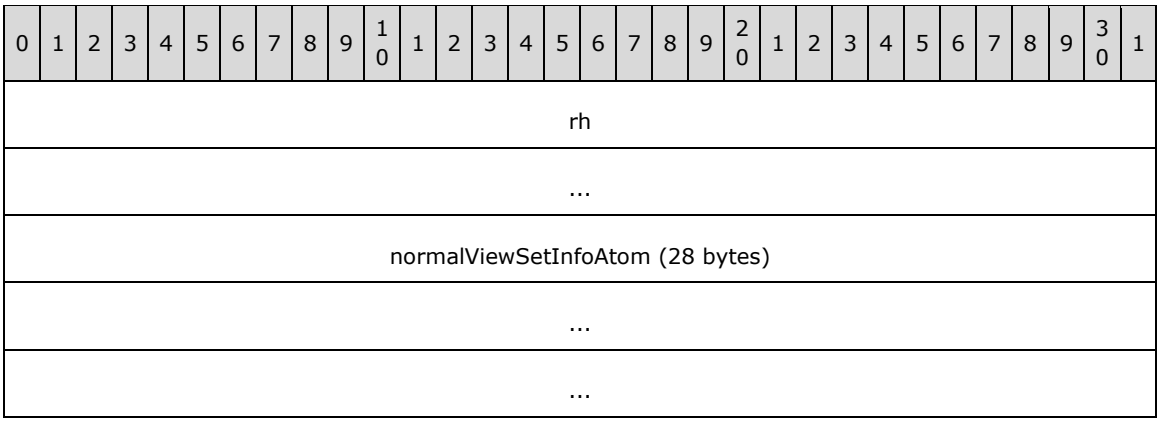
x (4 bytes): A signed integer that specifies horizontal grid spacing in **grid units**. It MUST be greater than or equal to 0x00005AB8 or 1 mm and less than or equal to 0x00120000 or 2 inches. It MUST be equal to **y**.

y (4 bytes): A signed integer that specifies vertical grid spacing in grid units. It MUST be greater than or equal to 0x00005AB8 or 1 mm and less than or equal to 0x00120000 or 2 inches. It MUST be equal to **x**.

2.4.21.2 NormalViewSetInfoContainer

Referenced by: DocInfoListSubContainerOrAtom

A container record that specifies display preferences for when a user interface shows the presentation in a manner optimized for the simultaneous display of all presentation slides, a specific presentation slide, and the text of the notes slide associated with that specific presentation slide.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be **RT_NormalViewSetInfo9** (section 2.13.24).
rh.Len	MUST be 0x0000001C.

normalViewSetInfoAtom (28 bytes): A NormalViewSetInfoAtom record that specifies the display preferences.

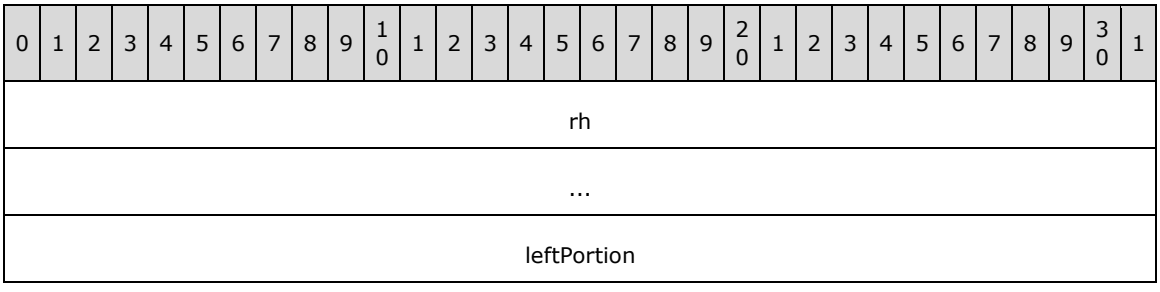
2.4.21.3 NormalViewSetInfoAtom

Referenced by: NormalViewSetInfoContainer

An atom record that specifies the appearance of different regions in a user interface view that consists of three content panes: a side content pane, a slide pane, and a notes text pane.

The side content pane can contain either thumbnail images of presentation slides in the presentation or a text outline of the presentation. It occupies the full height of the view and is separated from the slide pane and notes text pane by a vertical bar. It occupies the left edge of the view if the **fRightToLeft** field of the **DocumentAtom** record (section 2.4.2) is **FALSE** and the right edge of the view if the **fRightToLeft** field of the **DocumentAtom** record is **TRUE**.

The remainder of the view not occupied by the side content region is divided vertically by a horizontal bar. The slide pane displays a single presentation slide and is located above the horizontal bar. The notes text pane displays the text of the notes slide associated with the presentation slide and is located beneath the horizontal bar.



...					
topPortion					
...					
vertBarState	horizBarState	fPreferSingleSet	A	B	reserved

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_NormalViewSetInfo9Atom.
rh.recLen	MUST be 0x00000014

leftPortion (8 bytes): A **RatioStruct** structure (section 2.12.6) that specifies the width of the side content pane as a percentage of the view's width. The value of **leftPortion.numer** / **leftPortion.denom** MUST be greater than or equal to 0 and less than or equal to 1.

topPortion (8 bytes): A **RatioStruct** structure that specifies the height of the slide pane as a percentage of the view's height. The value of **topPortion.numer** / **topPortion.denom** MUST be greater than or equal to 0 and less than or equal to 1.

vertBarState (1 byte): A **NormalViewSetBarStates** enumeration that specifies the state of the vertical bar that separates the side content pane from the slide pane and notes text pane. If the value is BS_Minimized or BS_Maximized, the value of **leftPortion** MUST be ignored.

horizBarState (1 byte): A **NormalViewSetBarStates** enumeration that specifies the state of the horizontal bar that separates the slide pane from the notes text pane. If the value is BS_Minimized or BS_Maximized, the value of **topPortion** MUST be ignored.

fPreferSingleSet (1 byte): A **bool1** (section 2.2.2) that specifies whether the view consists of only the slide pane or all three panes. It MUST be a value from the following table:

Value	Meaning
0x01	The slide pane occupies the entire view.
0x00	All three panes exist in the view.

A - fHideThumbnails (1 bit): A bit that specifies the content of the side content pane. It MUST be a value from the following table.

Value	Meaning
0x1	The side content pane contains a text outline of the presentation.
0x0	The side content pane contains thumbnail images of the presentation slides.

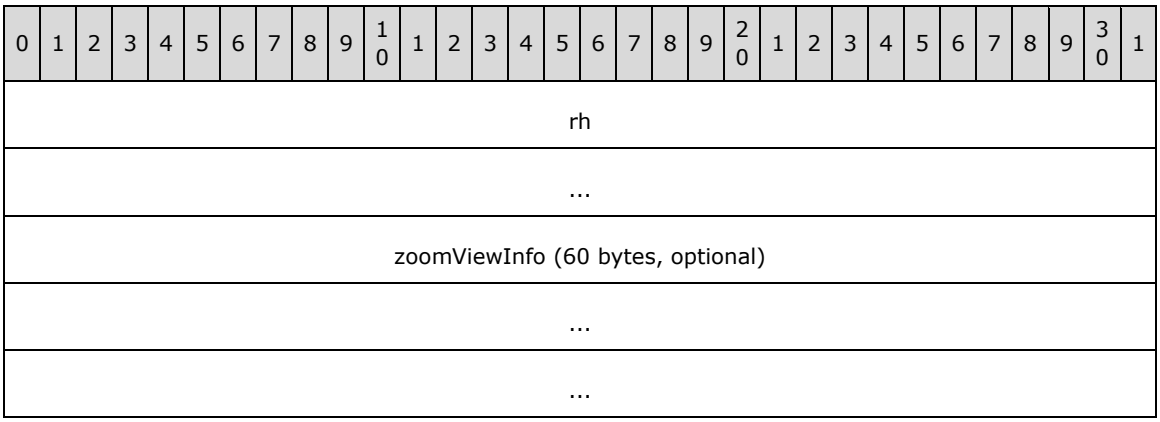
B - fBarSnapped (1 bit): A bit that specifies whether the vertical bar is snapped to specific positions when resized.

reserved (6 bits): MUST be zero and MUST be ignored.

2.4.21.4 NotesTextViewInfoContainer

Referenced by: *DocInfoListSubContainerOrAtom*

A container record that specifies display preferences for when a user interface shows the presentation in a manner optimized for the display of the text on the notes slides.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

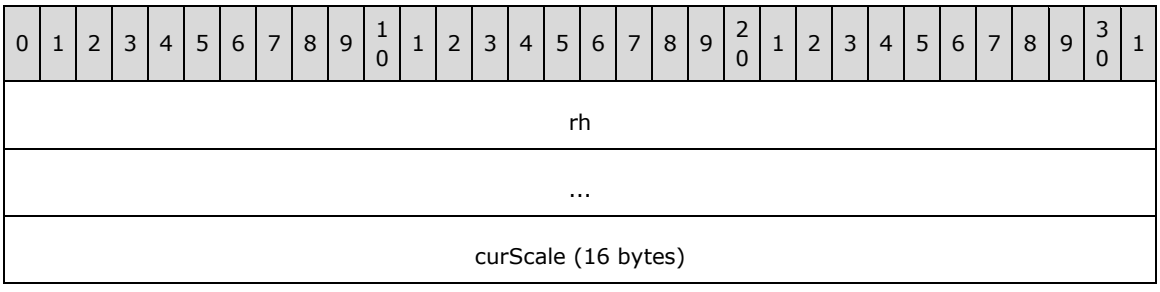
Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be **RT_NotesTextViewInfo9** (section 2.13.24).

zoomViewInfo (60 bytes): An optional ZoomViewInfoAtom record that specifies origin and scaling information.

2.4.21.5 ZoomViewInfoAtom

Referenced by: *NotesTextViewInfoContainer*, *NotesViewInfoContainer*, *SlideViewInfoContainer*

An atom record that specifies origin and scaling information for a view that can be zoomed beyond 100 percent.



...		
...		
unused1 (24 bytes)		
...		
...		
origin		
...		
fUseVarScale	fDraftMode	unused2

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_ViewInfoAtom.
rh.recLen	MUST be 0x00000034.

curScale (16 bytes): A ScalingStruct structure that specifies the scaling of content in the view. Sub fields are further specified in the following table.

Field	Meaning
curScale.x	Specifies scaling of the *x*-axis. The value of **curScale.x.numer** / **curScale.x.denom** MUST be greater than or equal to 0.10 and less than or equal to 4.0.
curScale.y	Specifies the scaling of the *y*-axis. The value of **curScale.y.numer** / **curScale.y.denom** MUST be equal to **curScale.x.numer** / **curScale.x.denom**.

unused1 (24 bytes): Undefined and MUST be ignored.

origin (8 bytes): A **PointStruct** structure (section 2.12.5) that specifies a position in master units, relative to the top-left corner of the full view, that is displayed in the top-left corner of the displayable view area.

fUseVarScale (1 byte): A **bool1** (section 2.2.2) that specifies how content is scaled. It MUST be a value from the following table.

Value	Meaning
0x00	Content is scaled as specified by **curScale**.
0x01	The scale varies with the size of the view such that

Value	Meaning
	the complete slide always occupies the entire view.

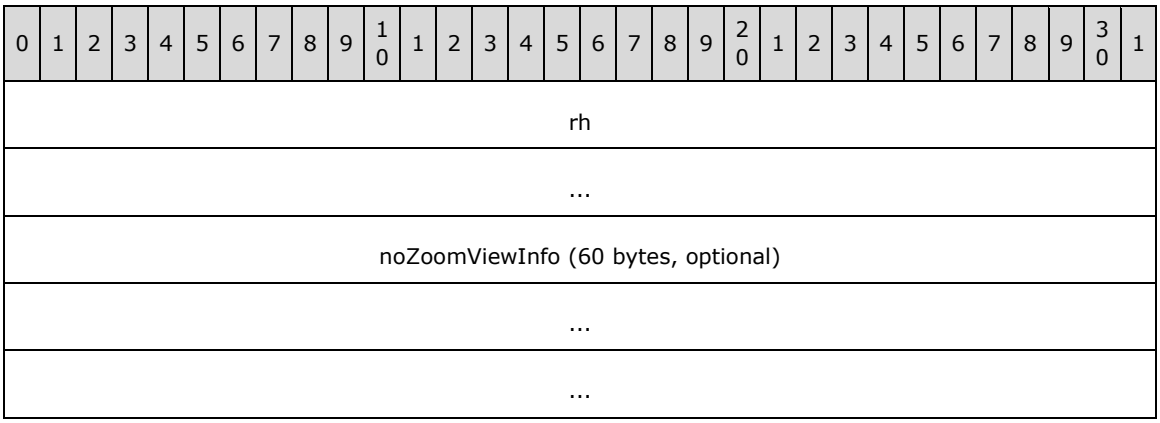
fDraftMode (1 byte): A **bool1** that specifies whether the view is displayed with less formatting.

unused2 (2 bytes): Undefined and MUST be ignored.

2.4.21.6 OutlineViewInfoContainer

Referenced by: DocInfoListSubContainerOrAtom

A container record that specifies display preferences for when a user interface shows the presentation in a manner optimized for the display of the text on the presentation slides.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

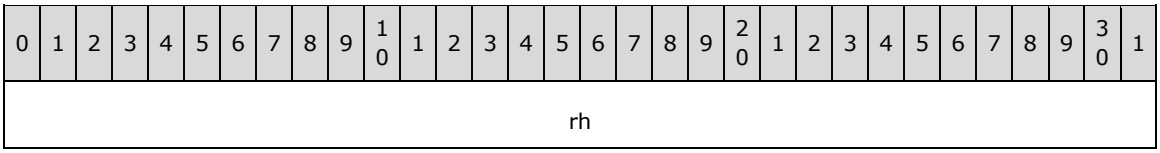
Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be **RT_OutlineViewInfo** (section 2.13.24).

noZoomViewInfo (60 bytes): An optional NoZoomViewInfoAtom record that specifies origin and scaling information.

2.4.21.7 NoZoomViewInfoAtom

Referenced by: OutlineViewInfoContainer, SorterViewInfoContainer

An atom record that specifies origin and scaling information for a view that cannot be zoomed beyond 100 percent.



...		
curScale (16 bytes)		
...		
...		
unused1 (24 bytes)		
...		
...		
origin		
...		
unused2	fDraftMode	unused3

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_ViewInfoAtom.
rh.recLen	MUST be 0x00000034.

curScale (16 bytes): A ScalingStruct structure that specifies the scaling of content in the view. Sub-fields are further specified in the following table.

Field	Meaning
curScale.x	Specifies scaling of the *x*-axis. The value of **curScale.x.numer** / **curScale.x.denom** MUST be greater than or equal to 0.20 and less than or equal to 1.0.
curScale.y	Specifies scaling of the *y*-axis. The value of **curScale.y.numer** / **curScale.y.denom** MUST be equal to **curScale.x.numer** / **curScale.x.denom**.

unused1 (24 bytes): Undefined and MUST be ignored.

origin (8 bytes): A **PointStruct** structure (section 2.12.5) that specifies a position in master units, relative to the top-left corner of the full view, that is displayed in the top-left corner of the displayable view area.

unused2 (1 byte): Undefined and MUST be ignored.

fDraftMode (1 byte): A **bool1** (section 2.2.2) that specifies whether the view is displayed with less formatting.

unused3 (2 bytes): Undefined and MUST be ignored.

2.4.21.8 SlideViewInfoInstance

Referenced by: DocInfoListSubContainerOrAtom

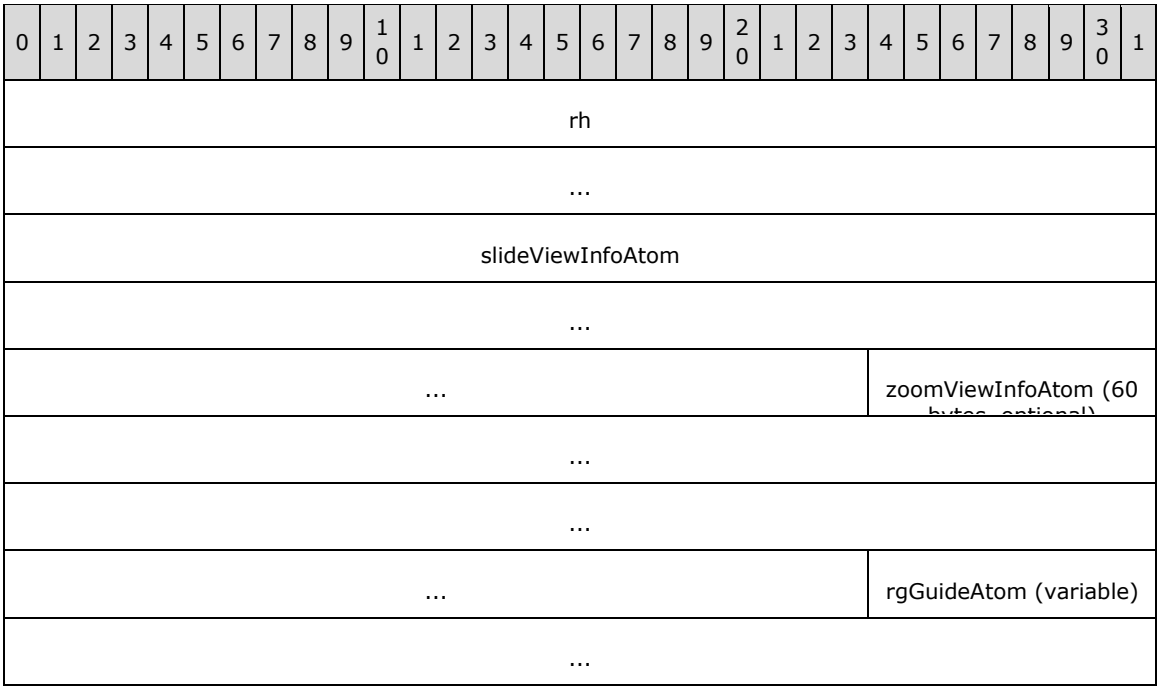
A variable type record whose type and meaning are dictated by the value of **rh.recInstance**, as specified in the following table.

Value	Meaning
0x000	A **SlideViewInfoContainer** record (section 2.4.21.9) that specifies display preferences for when a user interface shows the presentation in a manner optimized for the display of presentation slides.
0x001	A **NotesViewInfoContainer** record (section 2.4.21.12) that specifies display preferences for when a user interface shows the presentation in a manner optimized for the display of notes slides.

2.4.21.9 SlideViewInfoContainer

Referenced by: SlideViewInfoInstance

A container record that specifies display preferences for when a user interface shows the presentation in a manner optimized for the display of presentation slides.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.

Field	Meaning
rh.recType	MUST be **RT_SlideViewInfo** (section 2.13.24).

slideViewInfoAtom (11 bytes): A SlideViewInfoAtom record that specifies editing preferences for content positioning.

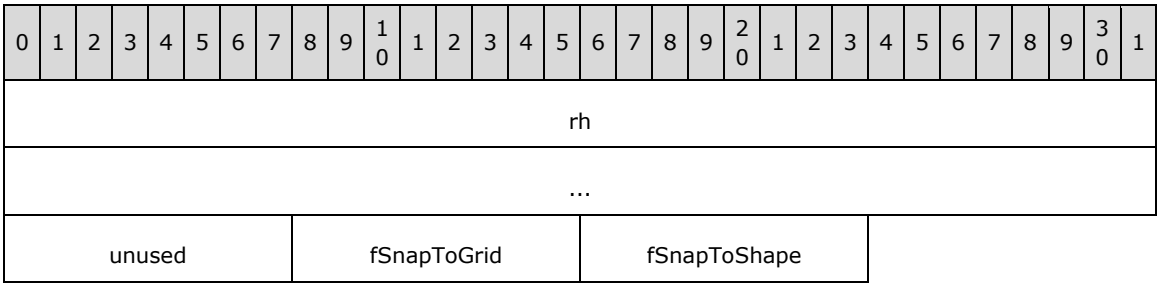
zoomViewInfoAtom (60 bytes): An optional ZoomViewInfoAtom record that specifies origin and scaling information.

rgGuideAtom (variable): An array of GuideAtom records that specifies **guides** for the slide view. It MUST NOT contain more than eight GuideAtom records with **type** equal to 0x00000000 (horizontal) and MUST NOT contain more than eight GuideAtom records with **type** equal to 0x00000001 (vertical). The array continues while **rh.recType** of the GuideAtom record is equal to RT_GuideAtom.

2.4.21.10 SlideViewInfoAtom

Referenced by: NotesViewInfoContainer, SlideViewInfoContainer

An atom record that specifies editing preferences for content positioning.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_SlideViewInfoAtom.
rh.recLen	MUST be 0x00000003.

unused (1 byte): Undefined and MUST be ignored.

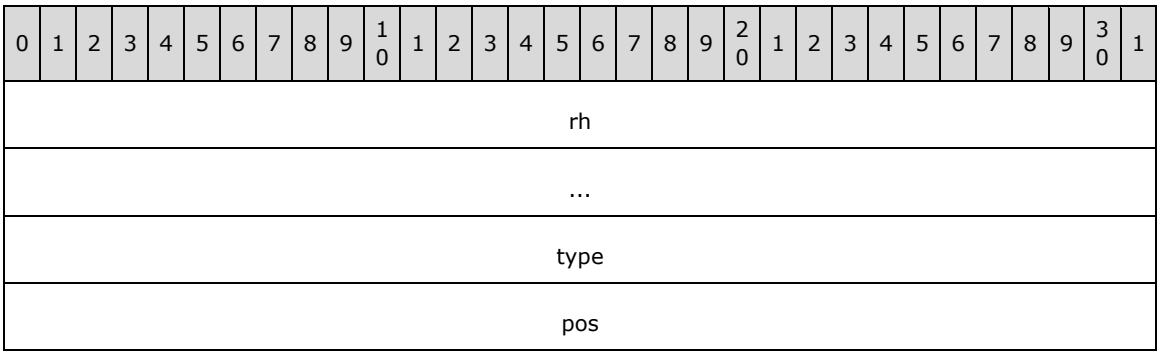
fSnapToGrid (1 byte): A **bool1** (section 2.2.2) that specifies an editing preference that the position of a shape aligns to the grid specified by the GridSpacing2.4.21.1Atom record.

fSnapToShape (1 byte): A **bool1** that specifies an editing preference that the position of a shape aligns to the position of other shapes.

2.4.21.11 GuideAtom

Referenced by: NotesViewInfoContainer, SlideViewInfoContainer

An atom record that specifies a guide. A guide can be used to align objects on a slide and to display visual positioning cues.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x007.
rh.recType	MUST be RT_GuideAtom.
rh.recLen	MUST be 0x00000008.

type (4 bytes): An **unsigned integer** that specifies whether the guide is horizontal or vertical. It MUST be a value from the following table.

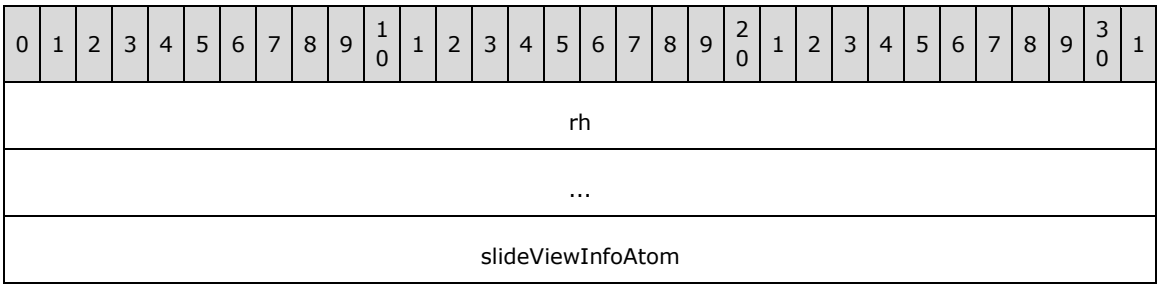
Value	Meaning
0x00000000	The guide is horizontal.
0x00000001	The guide is vertical.

pos (4 bytes): A **signed integer** that specifies the position of the guide in master units relative to the top-left corner of the slide. It MUST be greater than or equal to -15840 or -27.5 inches and less than or equal to 32255 or 56 inches. Typical values range from zero to slide height for a horizontal guide and from zero to slide width for a vertical guide.

2.4.21.12 NotesViewInfoContainer

Referenced by: SlideViewInfoInstance

A container record that specifies display preferences for when a user interface shows the presentation in a manner optimized for the display of notes slides.



...	
...	zoomViewInfoAtom (60 bytes, optional)
...	
...	
...	rgGuideAtom (variable)
...	

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be **RT_SlideViewInfo** (section 2.13.24).

slideViewInfoAtom (11 bytes): A SlideViewInfoAtom record that specifies editing preferences for content positioning.

zoomViewInfoAtom (60 bytes): An optional ZoomViewInfoAtom record that specifies origin and scaling information.

rgGuideAtom (variable): An array of GuideAtom records that specifies guides for the notes view. It MUST NOT contain more than eight GuideAtom records with **type** equal to 0x00000000 (horizontal) and MUST NOT contain more than eight GuideAtom records with **type** equal to 0x00000001 (vertical). The array continues while **rh.recType** of the GuideAtom record is equal to RT_GuideAtom.

2.4.21.13 SorterViewInfoContainer

Referenced by: DocInfoListSubContainerOrAtom

A container record that specifies display preferences for when a user interface shows the presentation in a manner optimized for the simultaneous display of multiple presentation slides.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
viewInfo (60 bytes, optional)																															
...																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be **RT_SorterViewInfo** (section 2.13.24).

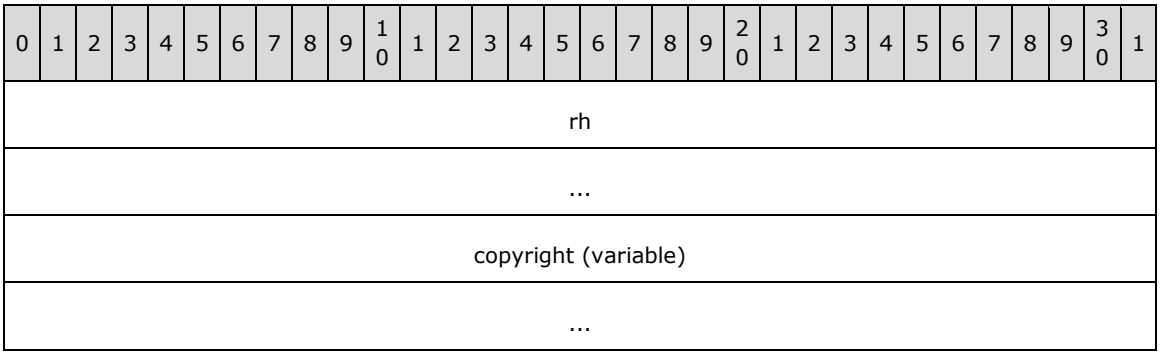
viewInfo (60 bytes): An optional NoZoomViewInfoAtom record that specifies origin and scaling information.

2.4.22 Summary Info Types

2.4.22.1 CopyrightAtom

Referenced by: PP10DocBinaryTagExtension

An atom record that specifies copyright information.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

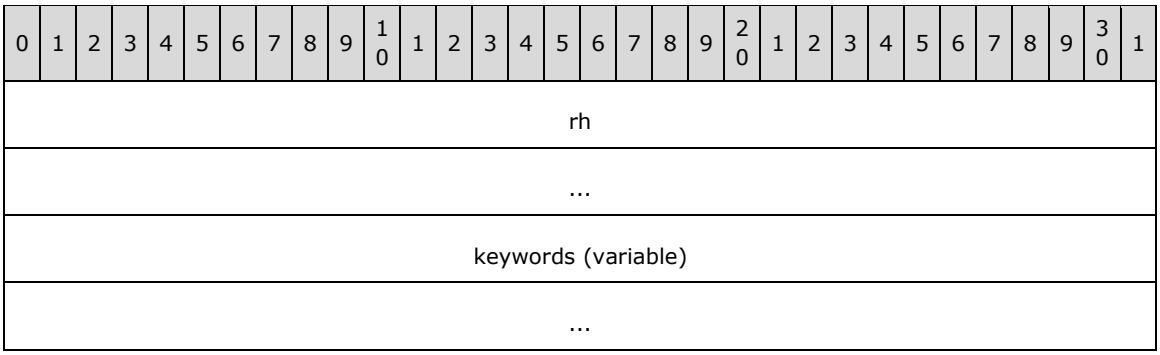
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 510.

copyright (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies copyright information. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.22.2 KeywordsAtom

Referenced by: PP10DocBinaryTagExtension

An atom record that specifies keyword information.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

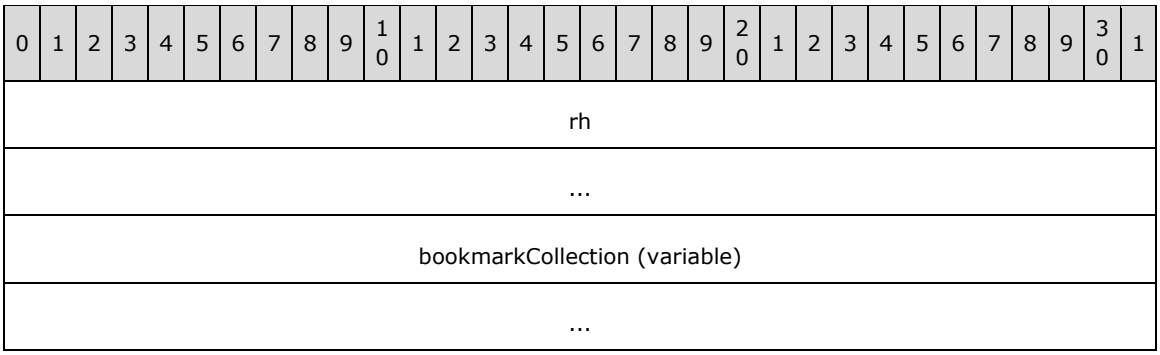
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x002.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 510.

keywords (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies keyword information. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.22.3 SummaryContainer

Referenced by: DocumentContainer

A container record that specifies bookmark information.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

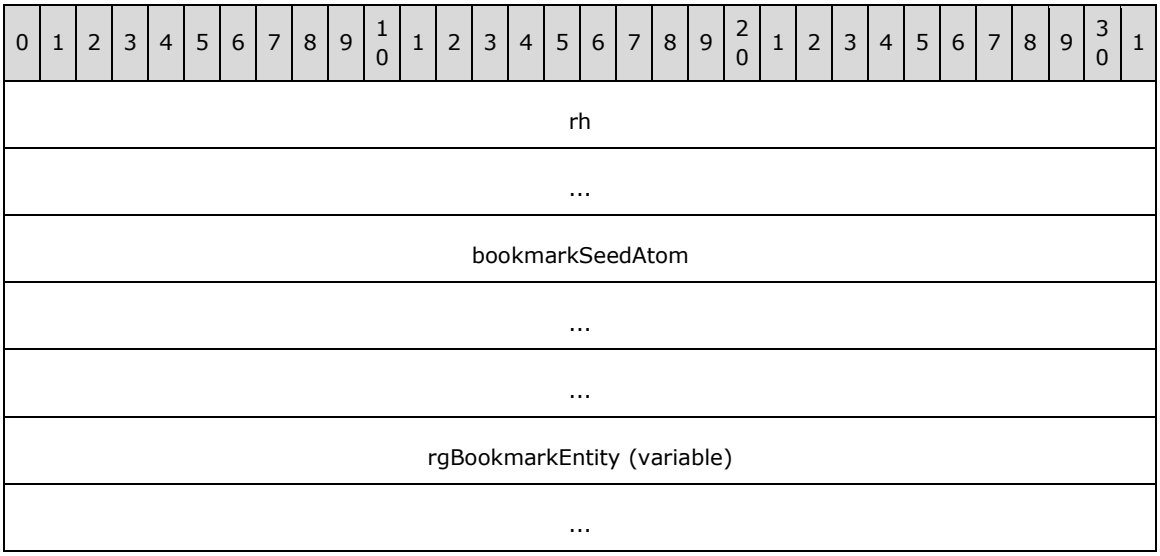
Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_Summary.

bookmarkCollection (variable): A BookmarkCollectionContainer record that specifies the bookmarks.

2.4.22.4 BookmarkCollectionContainer

Referenced by: SummaryContainer

A container record that specifies a collection of bookmarks.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_BookmarkCollection.

bookmarkSeedAtom (12 bytes): A BookmarkSeedAtom record that specifies the identifier to use when creating a new bookmark.

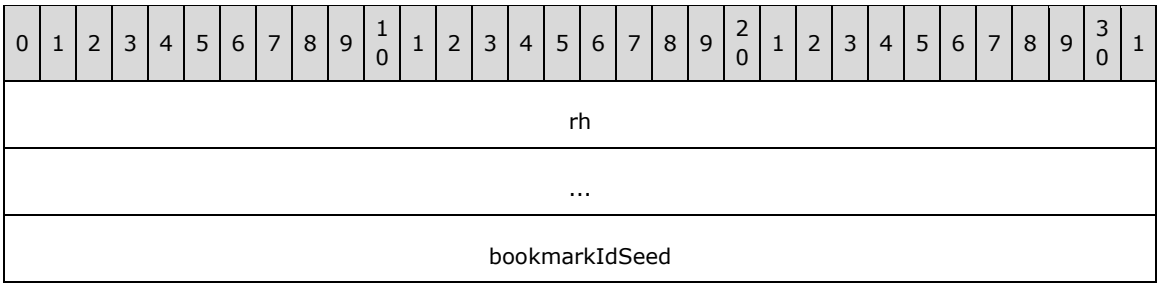
rgBookmarkEntity (variable): An array of BookmarkEntityAtomContainer records that specifies the bookmarks. The size, in bytes, of the array is specified by the following formula:

```
rh.recLen - 12
```

2.4.22.5 BookmarkSeedAtom

Referenced by: BookmarkCollectionContainer

An atom record that specifies the seed value to use when creating new bookmark identifiers.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

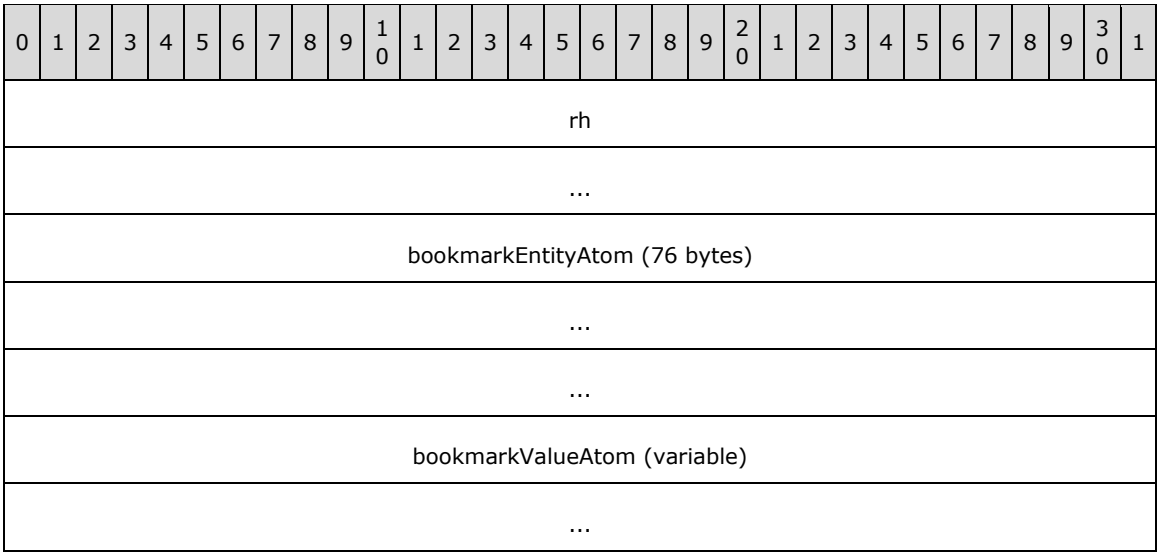
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x002.
rh.recType	MUST be RT_BookmarkSeedAtom.
rh.recLen	MUST be 0x00000004.

bookmarkIdSeed (4 bytes): An unsigned integer that specifies a seed for creating a new bookmark identifier. It MUST be greater than all existing bookmark identifier values specified by the **bookmarkID** field of the BookmarkEntityAtom records and the **bookmarkID** field of the TextBookMarkAtom records.

2.4.22.6 BookmarkEntityAtomContainer

Referenced by: BookmarkCollectionContainer

A container record that specifies information about a bookmark.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.

Field	Meaning
rh.recType	MUST be RT_BookmarkEntityAtom.

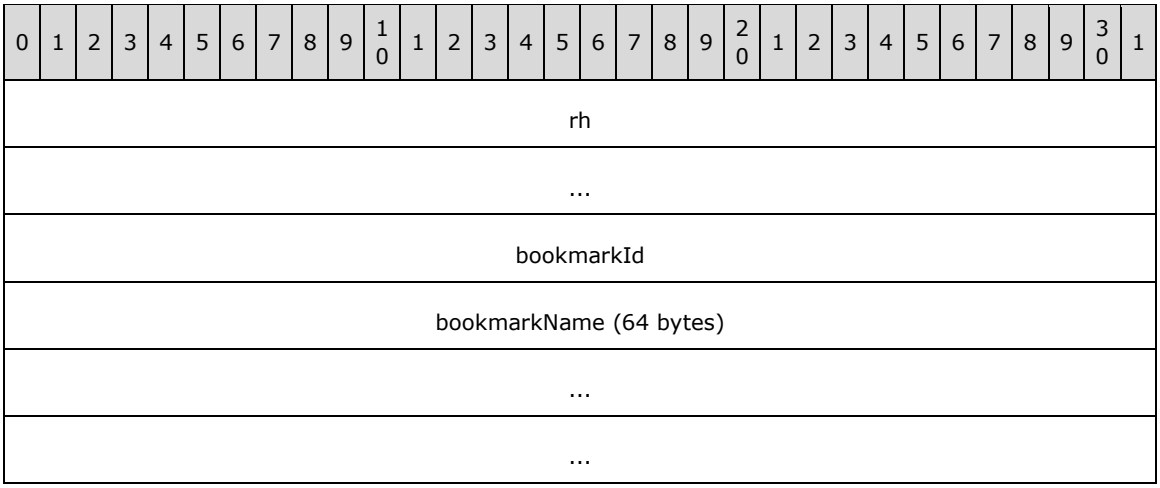
bookmarkEntityAtom (76 bytes): A BookMarkEntityAtom record that specifies how to link text to a bookmark.

bookmarkValueAtom (variable): A BookmarkValueAtom record that specifies the text value of the bookmark. This field MUST be the same as the text referred to by the associated TextBookMarkAtom record referred to by the **bookmarkId** field of the **bookmarkEntityAtom** in this record.

2.4.22.7 BookmarkEntityAtom

Referenced by: BookmarkEntityAtomContainer

An atom record that specifies information used to link the bookmark records in the text itself to the bookmarks in the Summary Information Stream.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_BookmarkEntityAtom.
rh.recLen	MUST be 0x00000044.

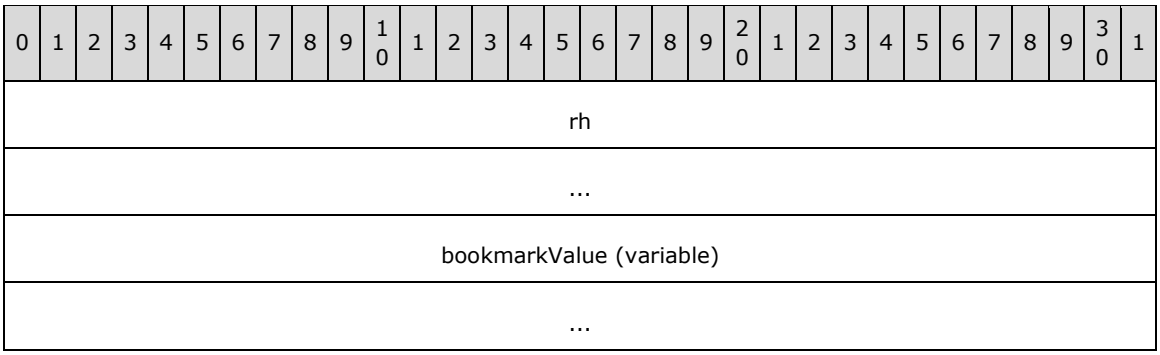
bookmarkId (4 bytes): An unsigned integer that specifies this bookmark identifier. It MUST be the same as the **bookmarkID** field of a TextBookMarkAtom record.

bookmarkName (64 bytes): A char2.2.4 that specifies the name of a bookmark. The name MUST NOT be empty. The name SHOULD be the same as one of the **PropertyIdentifierAndOffset.PropertyIdentifier** fields, as specified in [MS-OLEPS] section 2.19, in the Summary Information Stream.

2.4.22.8 BookmarkValueAtom

Referenced by: BookmarkEntityAtomContainer

An atom record that specifies the value of a bookmark.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be greater than zero and less than or equal to 510.

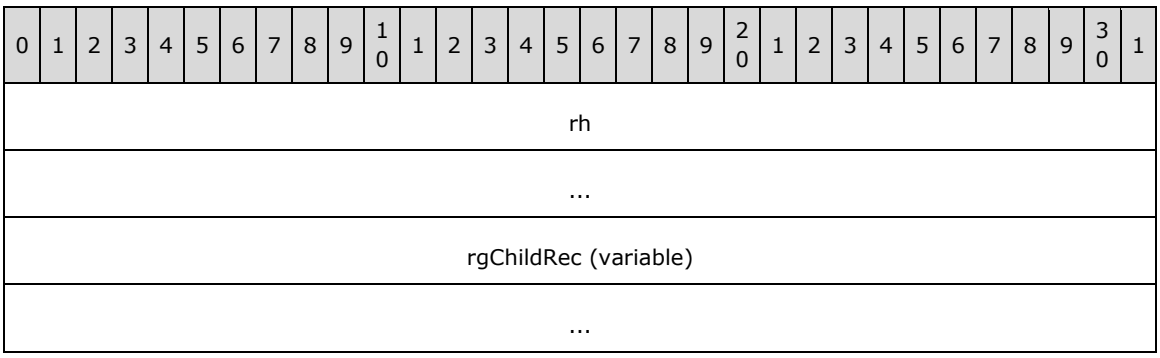
bookmarkValue (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the value of this bookmark. The length, in bytes, of the field is specified by **rh.recLen**.

2.4.23 Document Tag Info Types

2.4.23.1 DocProgTagsContainer

Referenced by: DocInfoListSubContainerOrAtom

A container record that specifies **programmable tags** with additional document data.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	SHOULD<15> be 0x000.
rh.recType	MUST be **RT_ProgTags** (section 2.13.24).

rgChildRec (variable): An array of DocProgTagsSubContainerOrAtom records that specifies the programmable tags. The size, in bytes, of the array is specified by **rh.recLen**. The array MUST NOT contain more than one of each of the following records: PP2.4.23.5DocBinaryTagExtension, PP2.4.23.6DocBinaryTagExtension, PP2.4.23.7DocBinaryTagExtension, PP2.4.23.8DocBinaryTagExtension.

2.4.23.2 DocProgTagsSubContainerOrAtom

Referenced by: DocProgTagsContainer

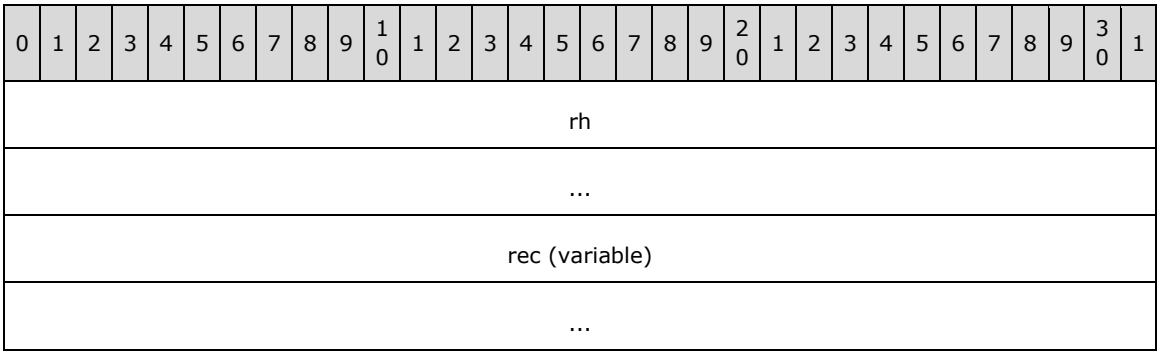
A variable type record whose type and meaning are dictated by the value of **rh.recType**, as specified in the following table.

Value	Meaning
RT_ProgStringTag	A ProgStringTagContainer record that specifies additional document data.
RT_ProgBinaryTag	A DocProgBinaryTagContainer record that specifies additional document data.

2.4.23.3 DocProgBinaryTagContainer

Referenced by: DocProgTagsSubContainerOrAtom

A container record that specifies programmable tags with additional binary document data.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_ProgBinaryTag.

rec (variable): A DocProgBinaryTagSubContainerOrAtom record that specifies additional document data.

2.4.23.4 DocProgBinaryTagSubContainerOrAtom

Referenced by: DocProgBinaryTagContainer

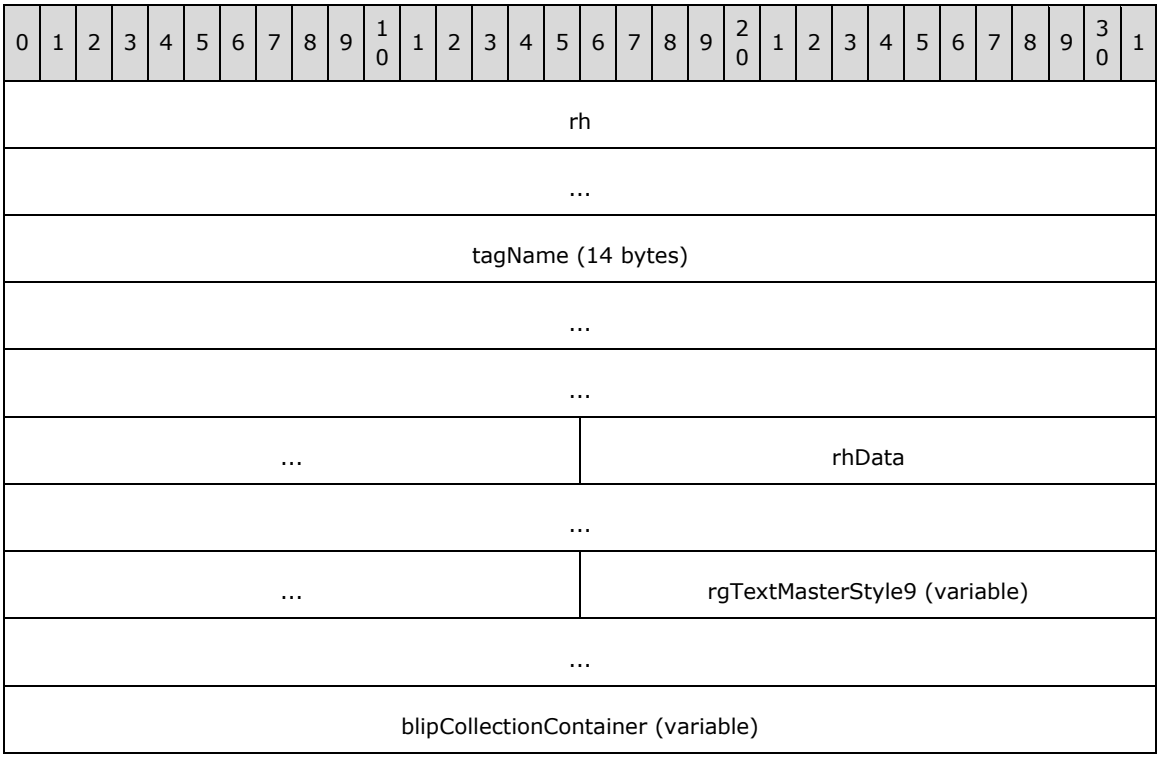
A variable type record whose type and meaning are dictated by the value of **tagNameAtom.tagName** for UnknownBinaryTag or by the value of **tagName** for PP2.4.23.5DocBinaryTagExtension, PP2.4.23.6DocBinaryTagExtension, PP2.4.23.7DocBinaryTagExtension, PP2.4.23.8DocBinaryTagExtension, as specified in the following table.

Value	Meaning
"___PPT9"	A PP2.4.23.5DocBinaryTagExtension record pair that specifies additional document data. It MAY<16> be ignored and MUST be preserved.
"___PPT10"	A PP2.4.23.6DocBinaryTagExtension record pair that specifies additional document data. It MAY<17> be ignored and MUST be preserved.
"___PPT11"	A PP2.4.23.7DocBinaryTagExtension record pair that specifies additional document data. It MAY<18> be ignored and MUST be preserved.
"___PPT12"	A PP2.4.23.8DocBinaryTagExtension record pair that specifies additional document data. It MAY<19> be ignored and MUST be preserved.
Any other value	An UnknownBinaryTag record pair that specifies additional document data. It MUST be ignored and MUST be preserved.

2.4.23.5 PP9DocBinaryTagExtension

Referenced by: DocProgBinaryTagSubContainerOrAtom

A pair of atom records that specifies a programmable tag with additional document data.



...
textDefaultsAtom (variable)
...
kinsokuContainer (variable)
...
rgExternalHyperlink9 (variable)
...
presAdvisorFlagsAtom (optional)
...
...
envelopeDataAtom (variable)
...
envelopeFlagsAtom (optional)
...
...
htmlDocInfoAtom (24 bytes, optional)
...
...
htmlPublishInfoAtom (variable)
...
rgBroadcastDocInfo9 (variable)
...
outlineTextPropsContainer (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the first record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be 0x0000000E.

tagName (14 bytes): A **PrintableUnicodeString** (section 2.2.23) that specifies the programmable tag name. It MUST be "___PPT9".

rhData (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the second record. Sub-fields are further specified in the following table:

Field	Meaning
rhData.recVer	MUST be 0x0.
rhData.recInstance	MUST be 0x000.
rhData.recType	MUST be RT_BinaryTagDataBlob.

rgTextMasterStyle9 (variable): An array of TextMasterStyle2.9.37Atom records that specifies additional character-level and paragraph-level formatting of main master slides. The array continues while **rh.recType** of the TextMasterStyle2.9.37Atom is equal to RT_TextMasterStyle9Atom.

blipCollectionContainer (variable): An optional **BlipCollection9Container** record (section 2.9.72) that specifies information about picture bullet points.

textDefaultsAtom (variable): An optional TextDefaults2.9.74Atom record that specifies additional default character-level and paragraph-level formatting.

kinsokuContainer (variable): An optional **Kinsoku9Container** (section 2.9.6) that specifies the user preferences for East Asian text line break settings.

rgExternalHyperlink9 (variable): An array of ExHyperlink2.10.21Container records that specifies additional information about hyperlinks. The array continues while **rh.recType** of the ExHyperlink2.10.21Container record is equal to RT_ExternalHyperlink9.

presAdvisorFlagsAtom (12 bytes): An optional PresAdvisorFlags2.4.6Atom record that specifies Presentation Assistant settings. It SHOULD<20> be ignored.

envelopeDataAtom (variable): An optional EnvelopeData2.11.4Atom record that specifies data for an **envelope**. It SHOULD<21> be ignored.

envelopeFlagsAtom (12 bytes): An optional EnvelopeFlags2.11.3Atom record that specifies information about an envelope. It SHOULD<22> be ignored.

htmlDocInfoAtom (24 bytes): An optional HTMLDocInfo2.4.18.1Atom record that specifies settings how to publish a document as a Web page.

htmlPublishInfoAtom (variable): An optional HTMLPublishInfo2.4.18.2Container record that contains additional information specifying how to publish a document as a Web page.

rgBroadcastDocInfo9 (variable): An array of BroadcastDocInfo2.4.17.1Container records that specifies settings for a presentation broadcast. The array continues while **rh.recType** of the BroadcastDocInfo2.4.17.1Container is equal to RT_BroadcastDocInfo9. It SHOULD<23> be ignored.

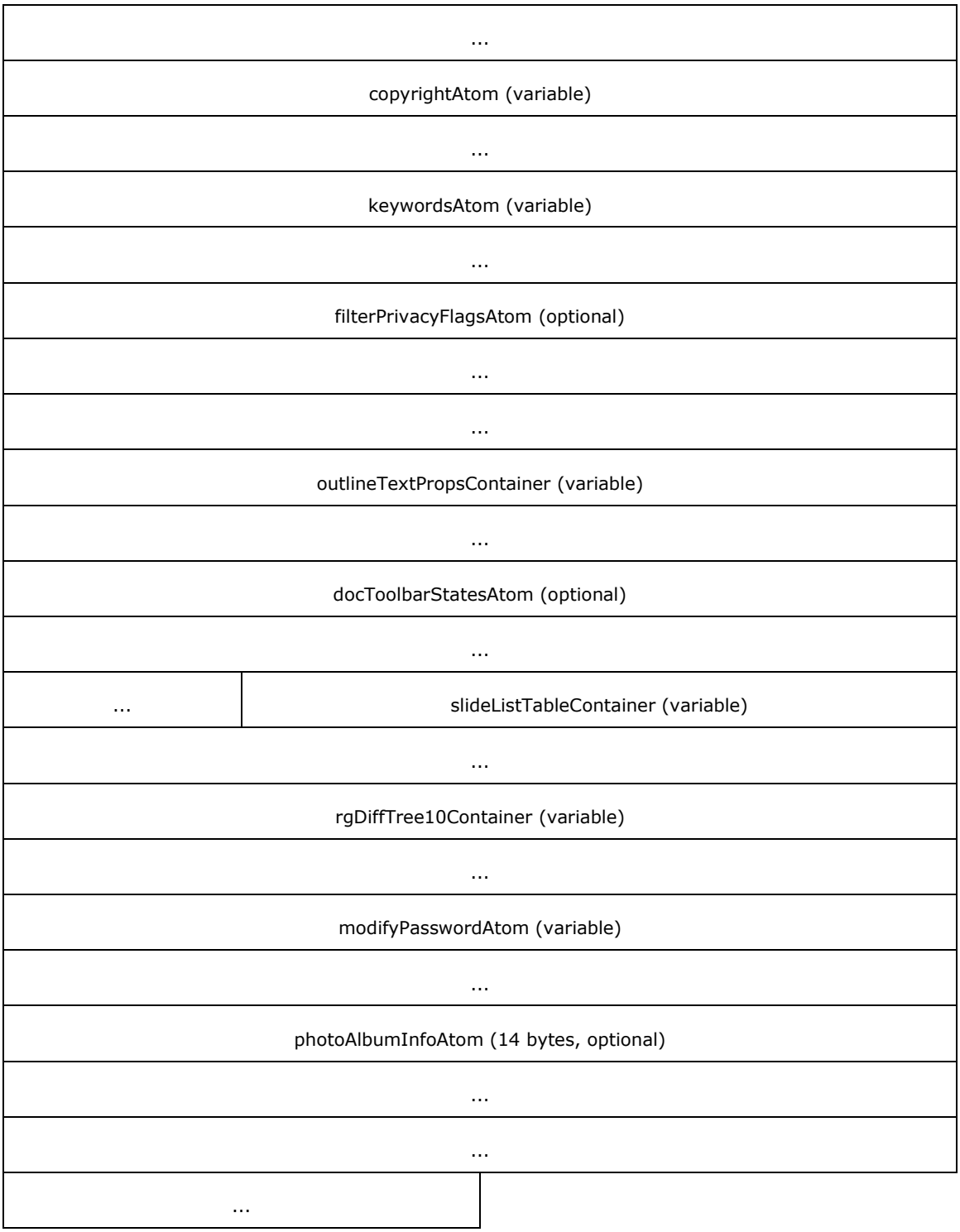
outlineTextPropsContainer (variable): An optional OutlineTextProps2.9.60Container record that specifies additional text properties for outline text.

2.4.23.6 PP10DocBinaryTagExtension

Referenced by: DocProgBinaryTagSubContainerOrAtom

A pair of atom records that specifies a programmable tag with additional document data.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
tagName (16 bytes)																															
...																															
...																															
rhData																															
...																															
fontCollectionContainer (variable)																															
...																															
rgTextMasterStyle10 (variable)																															
...																															
textDefaultsAtom (variable)																															
...																															
gridSpacingAtom (16 bytes, optional)																															
...																															
...																															
rgCommentIndex10 (variable)																															
...																															
fontEmbedFlagsAtom (optional)																															
...																															



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the first record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.

Field	Meaning
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be 0x00000010.

tagName (16 bytes): A **PrintableUnicodeString** (section 2.2.23) that specifies the programmable tag name. It MUST be "___PPT10".

rhData (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the second record. Sub-fields are further specified in the following table.

Field	Meaning
rhData.recVer	MUST be 0x0.
rhData.recInstance	MUST be 0x000.
rhData.recType	MUST be RT_BinaryTagDataBlob.

fontCollectionContainer (variable): An optional **FontCollection10Container** record (section 2.9.11) that specifies information about additional fonts in the presentation.

rgTextMasterStyle10 (variable): An array of TextMasterStyle2.9.39Atom records that specifies additional character-level and paragraph-level formatting of main master slides. The array continues while **rh.recType** of the TextMasterStyle2.9.39Atom record is equal to RT_TextMasterStyle10Atom.

textDefaultsAtom (variable): An optional TextDefaults2.9.75Atom record that specifies additional default character-level formatting.

gridSpacingAtom (16 bytes): A GridSpacing2.4.21.1Atom record that specifies spacing for a grid that can be used to align objects on a slide and to display positioning cues.

rgCommentIndex10 (variable): An array of CommentIndex2.4.19.1Container records that specifies information for presentation comments in the document. The array continues while **rh.recType** of the CommentIndex2.4.19.1Container item is equal to RT_CommentIndex10.

fontEmbedFlagsAtom (12 bytes): An optional FontEmbedFlags2.9.12Atom record that specifies how font data is embedded.

copyrightAtom (variable): An optional CopyrightAtom record that specifies copyright information.

keywordsAtom (variable): An optional KeywordsAtom record that specifies keyword information.

filterPrivacyFlagsAtom (12 bytes): An optional FilterPrivacyFlags2.4.8Atom record that specifies privacy settings.

outlineTextPropsContainer (variable): An optional OutlineTextProps2.9.63Container record that specifies additional text properties for outline text.

docToolbarStatesAtom (9 bytes): An optional DocToolbarStates2.4.20.1Atom record that specifies display options for toolbars. It SHOULD<24> be ignored and SHOULD<25> be omitted.

slideListTableContainer (variable): An optional SlideListTable2.4.20.2Container record that specifies additional data about slides in the document. It SHOULD<26> be ignored and SHOULD<27> be omitted.

rgDiffTree10Container (variable): An optional array of DiffTree2.4.20.5Container. The array continues while **rh.recType** of the DiffTree2.4.20.5Container item is equal to RT_DiffTree10. The

array specifies the names of reviewers and how to display the changes of the document made by those reviewers. It SHOULD<28> be ignored and SHOULD<29> be omitted.

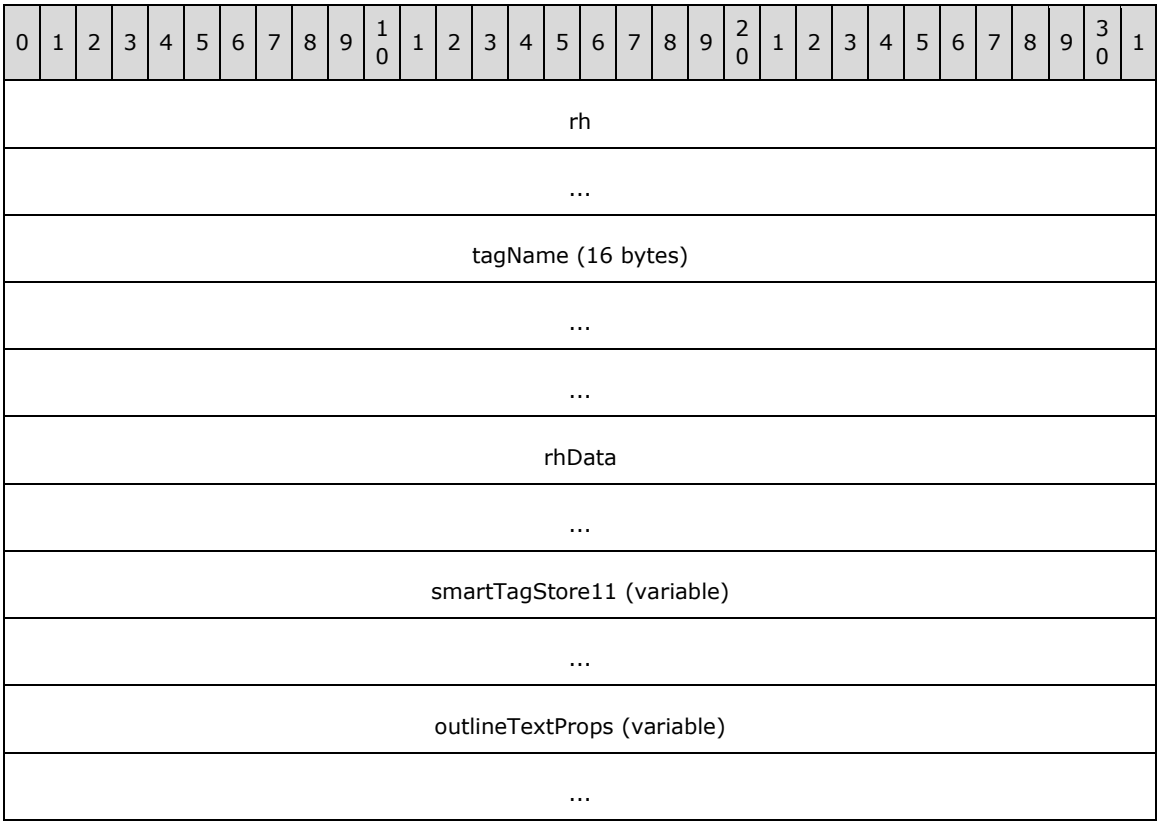
modifyPasswordAtom (variable): An optional ModifyPasswordAtom record that specifies a password used to modify the document.

photoAlbumInfoAtom (14 bytes): An optional PhotoAlbumInfo2.4.9Atom record that specifies user preferences for how to display a presentation as a photo album.

2.4.23.7 PP11DocBinaryTagExtension

Referenced by: DocProgBinaryTagSubContainerOrAtom

A pair of atom records that specifies a programmable tag with additional document data.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the first record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be 0x00000010.

tagName (16 bytes): A **PrintableUnicodeString** (section 2.2.23) that specifies the programmable tag name. It MUST be "___PPT11".

rhData (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the second record. Sub-fields are further specified in the following table.

Field	Meaning
rhData.recVer	MUST be 0x0.
rhData.recInstance	MUST be 0x000.
rhData.recType	MUST be RT_BinaryTagDataBlob.

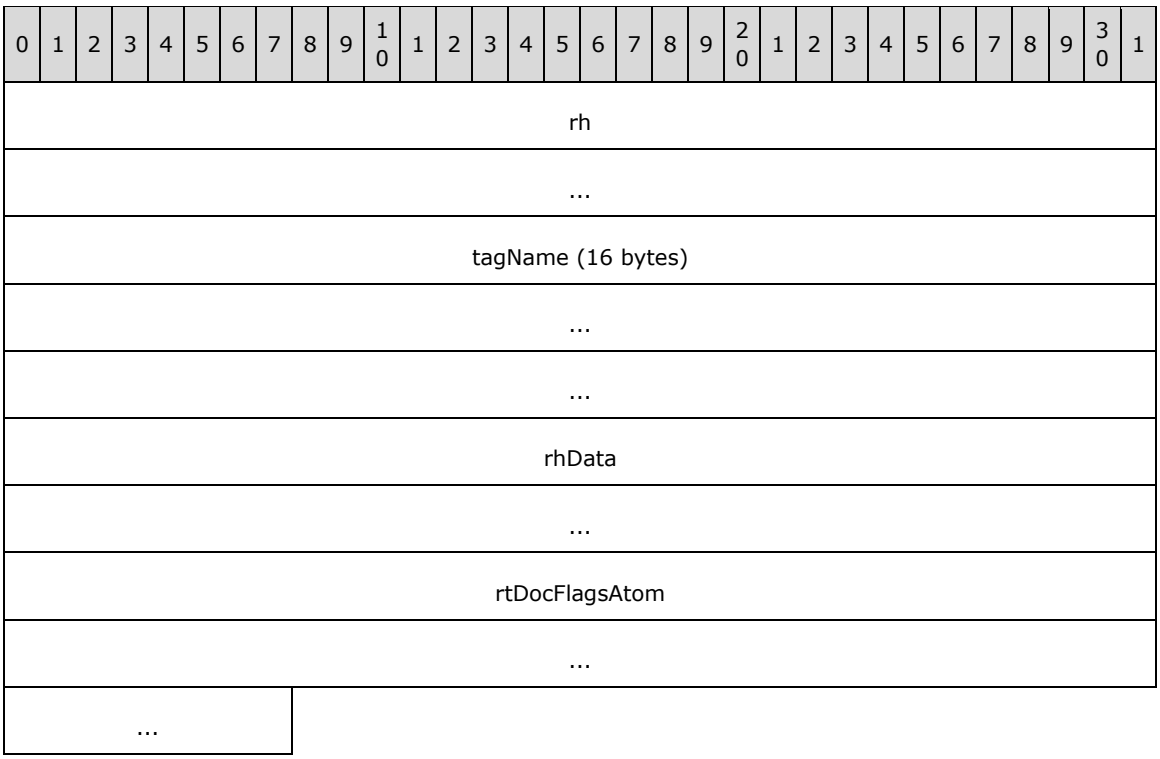
smartTagStore11 (variable): An optional **SmartTagStore11Container** record (section 2.11.28) that specifies smart tag data.

outlineTextProps (variable): An optional OutlineTextProps2.9.65Container record that specifies outline text data.

2.4.23.8 PP12DocBinaryTagExtension

Referenced by: DocProgBinaryTagSubContainerOrAtom

A pair of atom records that specifies a programmable tag with additional document data.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the first record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be 0x00000010.

tagName (16 bytes): A **PrintableUnicodeString** (section 2.2.23) that specifies the programmable tag name. It MUST be "___PPT12".

rhData (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the second record. Sub-fields are further specified in the following table.

Field	Meaning
rhData.recVer	MUST be 0x0.
rhData.recInstance	MUST be 0x000.
rhData.recType	MUST be RT_BinaryTagDataBlob.

rtDocFlagsAtom (9 bytes): An optional RoundTripDocFlags2.11.14Atom record that specifies document level flags. It SHOULD<30> be omitted.

2.5 Slide Types

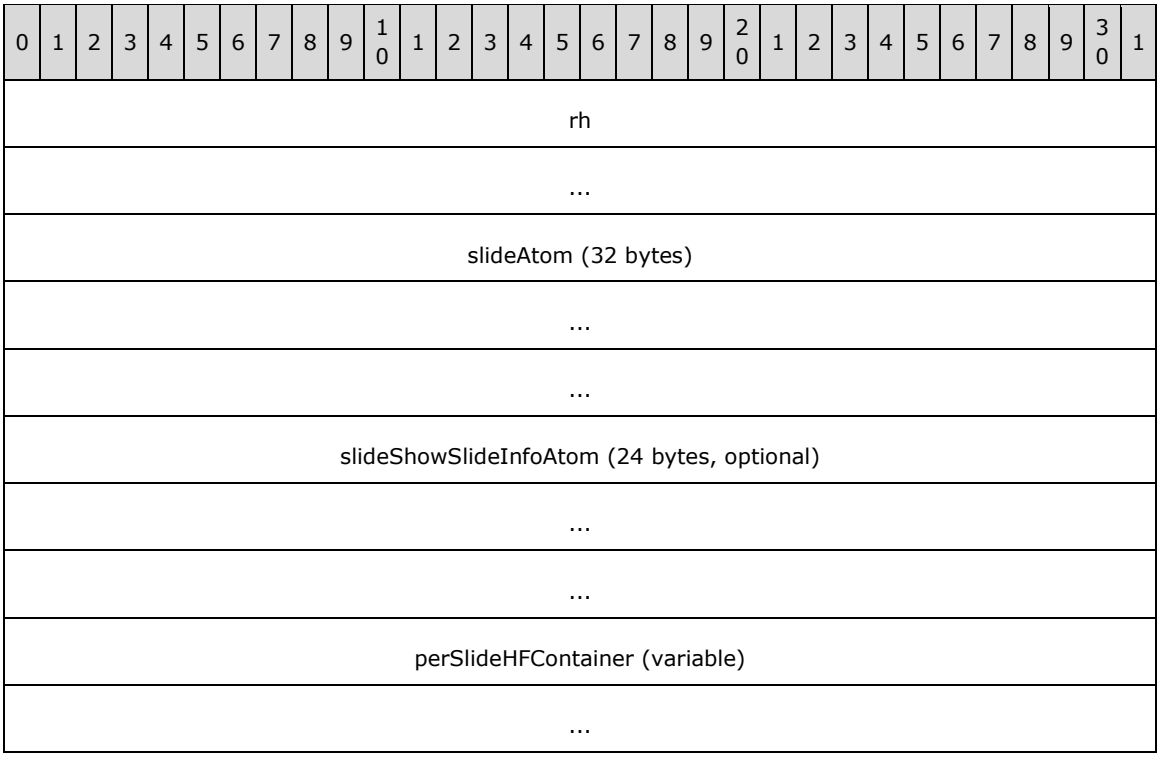
2.5.1 SlideContainer

Referenced by: MasterOrSlideContainer

A container record that specifies a presentation slide or title master slide.

Let the *corresponding master slide* be the **MainMasterContainer** record (section 2.5.3) or **SlideContainer** record specified by **slideAtom.masterIdRef**.

Let the *corresponding notes slide* be the **NotesContainer** record (section 2.5.6) specified by **slideAtom.notesIdRef**.



rtSlideSyncInfo12 (variable)
...
drawing (variable)
...
slideSchemeColorSchemeAtom (40 bytes)
...
...
slideNameAtom (variable)
...
slideProgTagsContainer (variable)
...
rgRoundTripSlide (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_Slide.

slideAtom (32 bytes): A SlideAtom record that specifies information specific to this slide.

slideShowSlideInfoAtom (24 bytes): An optional SlideShowSlideInfoAtom record that specifies slide transition information.

perSlideHFContainer (variable): An optional PerSlideHeadersFootersContainer record that specifies header and footer information for this slide. It SHOULD<31> be preserved.

rtSlideSyncInfo12 (variable): An optional RoundTripSlideSyncInfo2.11.23Container record that specifies round-trip information. It SHOULD<32> be ignored and SHOULD<33> be preserved.

drawing (variable): A **DrawingContainer** (section 2.5.13) that specifies drawing information for this slide.

slideSchemeColorSchemeAtom (40 bytes): A SlideSchemeColorSchemeAtom record that specifies the **color scheme** for this slide. If **slideAtom.slideFlags.fMasterScheme** is set, then the SlideSchemeColorSchemeAtom record contained by the *corresponding master slide* is used instead.

slideNameAtom (variable): An optional SlideNameAtom record that specifies the name of this slide. It SHOULD<34> be preserved.

slideProgTagsContainer (variable): An optional SlideProgTagsContainer record that specifies a list of programmable tags.

rgRoundTripSlide (variable): An array of RoundTripSlideRecord records that specifies round-trip information. The array continues while **rh.recType** of the RoundTripSlideRecord item is equal to one of the following record types: RT_RoundTripTheme12Atom, RT_RoundTripColorMapping12Atom, RT_RoundTripCompositeMasterId12Atom, RT_RoundTripSlideSyncInfo12, RT_RoundTripAnimationHashAtom12Atom, RT_RoundTripAnimationAtom12Atom, or RT_RoundTripContentMasterId12Atom. Each record type MUST NOT appear more than once.

2.5.2 RoundTripSlideRecord

Referenced by: SlideContainer

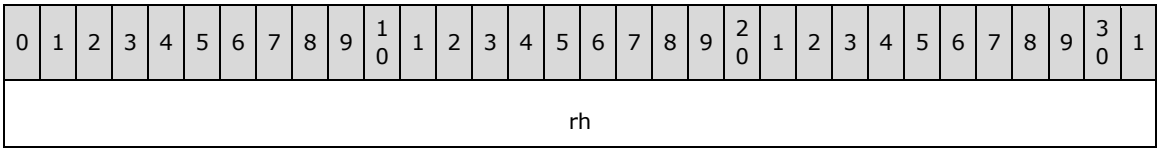
A variable type record whose type and meaning are dictated by the value of **rh.recType**, as specified in the following table.

Value	Meaning
RT_RoundTripTheme12Atom	A RoundTripThemeAtom record that specifies round-trip information. It SHOULD<35> be ignored and SHOULD<36> be preserved.
RT_RoundTripColorMapping12Atom	A RoundTripColorMappingAtom record that specifies round-trip information. It SHOULD<37> be ignored and SHOULD<38> be preserved.
RT_RoundTripCompositeMasterId12Atom	A RoundTripCompositeMasterId2.11.10Atom record that specifies round-trip information. It SHOULD<39> be ignored and SHOULD<40> be preserved.
RT_RoundTripSlideSyncInfo12	A RoundTripSlideSyncInfo2.11.23Container record that specifies round-trip information. It SHOULD<41> be ignored and SHOULD<42> be preserved.
RT_RoundTripAnimationHashAtom12Atom	A RoundTripAnimationHashAtom record that specifies round-trip information. It SHOULD<43> be ignored and SHOULD<44> be preserved.
RT_RoundTripAnimationAtom12Atom	A RoundTripAnimationAtom record that specifies round-trip information. It SHOULD<45> be ignored and SHOULD<46> be preserved.
RT_RoundTripContentMasterId12Atom	A RoundTripContentMasterId2.11.11Atom record that specifies round-trip information. It SHOULD<47> be ignored and SHOULD<48> be preserved.

2.5.3 MainMasterContainer

Referenced by: MasterOrSlideContainer

A container record that specifies a main master slide.



...
slideAtom (32 bytes)
...
...
rgSchemeListElementColorScheme (variable)
...
rgTextMasterStyle (variable)
...
roundTripOArtTextStyles12Atom (variable)
...
slideShowSlideInfoAtom (24 bytes, optional)
...
...
perSlideHeadersFootersContainer (variable)
...
drawing (variable)
...
slideSchemeColorSchemeAtom (40 bytes)
...
...
slideNameAtom (variable)
...
slideProgTagsContainer (variable)
...
rgRoundTripMainMaster (variable)

...
templateNameAtom (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_MainMaster.

slideAtom (32 bytes): A SlideAtom record that specifies slide-specific information.

rgSchemeListElementColorScheme (variable): An array of SchemeListElementColorSchemeAtom record that specifies a list of color schemes. The array continues while the **rh.recType** field of each SchemeListElementColorSchemeAtom item is equal to RT_ColorSchemeAtom.

rgTextMasterStyle (variable): An array of TextMasterStyleAtom record that specifies text formatting for this main master slide. It MUST contain at least one item with **rh.recInstance** equal to 0x000 (title placeholder) and at least one item with **rh.recInstance** equal to 0x001 (body placeholder). If this **MainMasterContainer** record is referenced by the first **MasterPersistAtom** record (section 2.4.14.2) contained within the **MasterListWithTextContainer** record (section 2.4.14.1), this array MUST also contain at least one item with **rh.recInstance** equal to 0x002 (notes placeholder). The array continues while the **rh.recType** field of each TextMasterStyleAtom item is equal to RT_TextMasterStyleAtom.

roundTripOArtTextStyles12Atom (variable): An optional RoundTripOArtTextStyles2.11.19Atom record that specifies round-trip information. It SHOULD<49> be ignored and SHOULD<50> be preserved.

slideShowSlideInfoAtom (24 bytes): An optional SlideShowSlideInfoAtom record that specifies slide show information for this main master slide.

perSlideHeadersFootersContainer (variable): An optional PerSlideHeadersFootersContainer record that specifies header and footer information for this main master slide. It SHOULD<51> be preserved.

drawing (variable): A **DrawingContainer** record (section 2.5.13) that specifies drawing information for this main master slide.

slideSchemeColorSchemeAtom (40 bytes): A SlideSchemeColorSchemeAtom record that specifies the color scheme for this main master slide.

slideNameAtom (variable): An optional SlideNameAtom record that specifies the name of this main master slide. It SHOULD<52> be preserved.

slideProgTagsContainer (variable): An optional SlideProgTagsContainer record that specifies a list of programmable tags.

rgRoundTripMainMaster (variable): An array of RoundTripMainMasterRecord records that specifies additional data for this main master slide. The array continues while **rh.recType** of the RoundTripMainMasterRecord item is equal to one of the following record types:

RT_RoundTripOriginalMainMasterId12Atom, RT_RoundTripTheme12Atom, RT_RoundTripColorMapping12Atom, RT_RoundTripContentMasterInfo12Atom, RT_RoundTripOArtTextStyles12Atom, RT_RoundTripAnimationHashAtom12Atom, RT_RoundTripAnimationAtom12Atom, or RT_RoundTripCompositeMasterId12Atom. Each record type MUST NOT appear more than once, except for the RT_RoundTripContentMasterInfo12Atom record type.

templateNameAtom (variable): An optional TemplateNameAtom record that specifies the design name of this main master slide.

2.5.4 RoundTripMainMasterRecord

Referenced by: MainMasterContainer

A variable type record whose type and meaning is dictated by the value of **rh.recType**, as specified in the following table.

Value	Meaning
RT_RoundTripOriginalMainMasterId12Atom	A RoundTripOriginalMainMasterId2.11.20Atom record that specifies round-trip information. It SHOULD<53> be ignored and SHOULD<54> be preserved.
RT_RoundTripTheme12Atom	A RoundTripThemeAtom record that specifies round-trip information. It SHOULD<55> be ignored and SHOULD<56> be preserved.
RT_RoundTripColorMapping12Atom	A RoundTripColorMappingAtom record that specifies round-trip information. It SHOULD<57> be ignored and SHOULD<58> be preserved.
RT_RoundTripContentMasterInfo12Atom	A RoundTripContentMasterInfo2.11.12Atom record that specifies round-trip information. It SHOULD<59> be ignored and SHOULD<60> be preserved.
RT_RoundTripOArtTextStyles12Atom	A RoundTripOArtTextStyles2.11.19Atom record that specifies round-trip information. It SHOULD<61> be ignored and SHOULD<62> be preserved.
RT_RoundTripAnimationHashAtom12Atom	A RoundTripAnimationHashAtom record that specifies round-trip information. It SHOULD<63> be ignored and SHOULD<64> be preserved.
RT_RoundTripAnimationAtom12Atom	A RoundTripAnimationAtom record that specifies round-trip information. It SHOULD<65> be ignored and SHOULD<66> be preserved.
RT_RoundTripCompositeMasterId12Atom	A RoundTripCompositeMasterId2.11.10Atom record that specifies round-trip information. It SHOULD<67> be ignored and SHOULD<68> be preserved.

2.5.5 MasterOrSlideContainer

A variable type record whose type and meaning is dictated by the value of **rh.recType**, as specified in the following table.

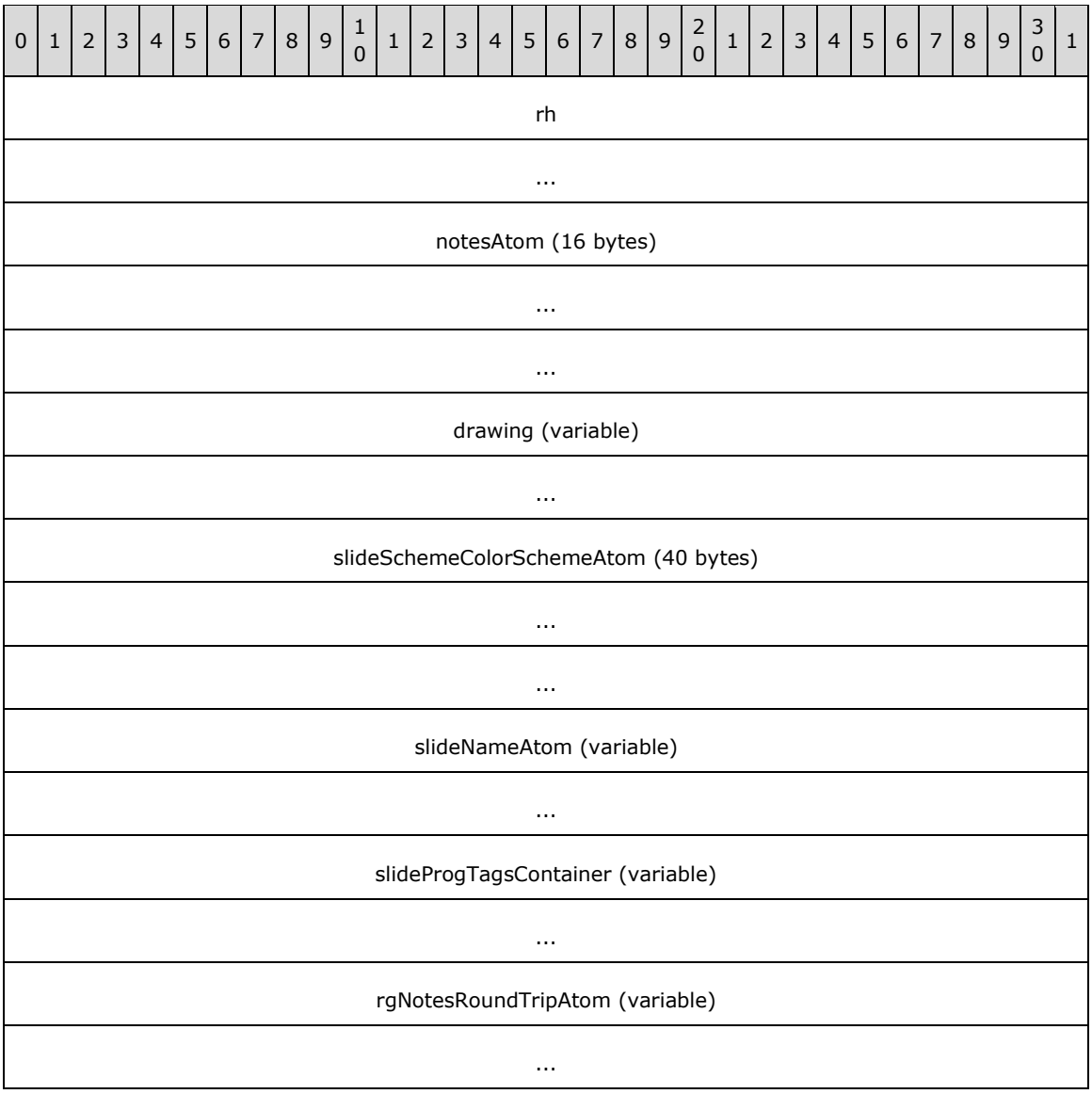
Value	Meaning
RT_Slide	A **SlideContainer** record (section 2.5.1) that specifies a title master slide.

Value	Meaning
RT_MainMaster	A **MainMasterContainer** record (section 2.5.3) that specifies a main master slide.

2.5.6 NotesContainer

A container record that specifies a notes slide or a notes master slide.

Let the *corresponding notes master* be specified by the **NotesContainer** record specified by the **notesMasterPersistIdRef** field of the **DocumentAtom** record (section 2.4.2).



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.

Field	Meaning
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_Notes.

notesAtom (16 bytes): A NotesAtom record that specifies data for this notes slide or notes master slide.

drawing (variable): A **DrawingContainer** record (section 2.5.13) that specifies the arrangement of content on this notes slide or notes master slide.

slideSchemeColorSchemeAtom (40 bytes): A SlideSchemeColorSchemeAtom record that specifies a color scheme for this notes slide or notes master slide. If **notesAtom.slideFlags.fMasterScheme** is set, the SlideSchemeColorSchemeAtom record contained by the *corresponding notes master* is used instead.

slideNameAtom (variable): An optional SlideNameAtom record that specifies a name for this notes slide or notes master slide. It SHOULD<69> be preserved.

slideProgTagsContainer (variable): An optional SlideProgTagsContainer record that specifies a list of programmable tags.

rgNotesRoundTripAtom (variable): An array of NotesRoundTripAtom records that specifies additional data for this notes slide or notes master slide. The array continues while **rh.recType** of the NotesRoundTripAtom item is equal to one of the following record types: RT_RoundTripTheme12Atom, RT_RoundTripColorMapping12Atom, or RT_RoundTripNotesMasterTextStyles12Atom. Each record type MUST NOT appear more than once.

2.5.7 NotesRoundTripAtom

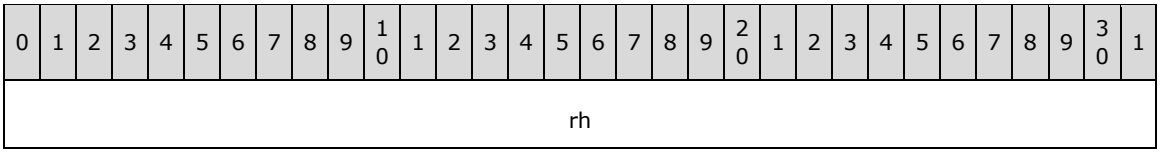
Referenced by: NotesContainer

A variable type record whose type and meaning is dictated by the value of **rh.recType**, as specified in the following table.

Value	Meaning
RT_RoundTripTheme12Atom	A RoundTripThemeAtom record that specifies round-trip information. It SHOULD<70> be ignored and SHOULD<71> be preserved.
RT_RoundTripColorMapping12Atom	A RoundTripColorMappingAtom record that specifies round-trip information. It SHOULD<72> be ignored and SHOULD<73> be preserved.
RT_RoundTripNotesMasterTextStyles12Atom	A RoundTripNotesMasterTextStyles2.11.18Atom record that specifies round-trip information. It SHOULD<74> be ignored and SHOULD<75> be preserved.

2.5.8 HandoutContainer

A container record that specifies the handout master slide.



...
drawing (variable)
...
slideSchemeColorSchemeAtom (40 bytes)
...
...
slideNameAtom (variable)
...
slideProgTagsContainer (variable)
...
rgHandoutRoundTripAtom (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_Handout.

drawing (variable): A **DrawingContainer** record (section 2.5.13) that specifies the arrangement of content on the handout master slide.

slideSchemeColorSchemeAtom (40 bytes): A SlideSchemeColorSchemeAtom record that specifies the color scheme for the handout master slide.

slideNameAtom (variable): An optional SlideNameAtom record that specifies the name for the handout master slide. It SHOULD<76> be preserved.

slideProgTagsContainer (variable): An optional SlideProgTagsContainer record that specifies a list of programmable tags.

rgHandoutRoundTripAtom (variable): An array of HandoutRoundTripAtom records that specifies round-trip information. The array continues while **rh.recType** of the HandoutRoundTripAtom item is equal to one of the following record types: RT_RoundTripTheme12Atom or RT_RoundTripColorMapping12Atom. Each record type MUST NOT appear more than once.

2.5.9 HandoutRoundTripAtom

Referenced by: HandoutContainer

A variable type record whose type and meaning are dictated by the value of **rh.recType**, as specified in the following table.

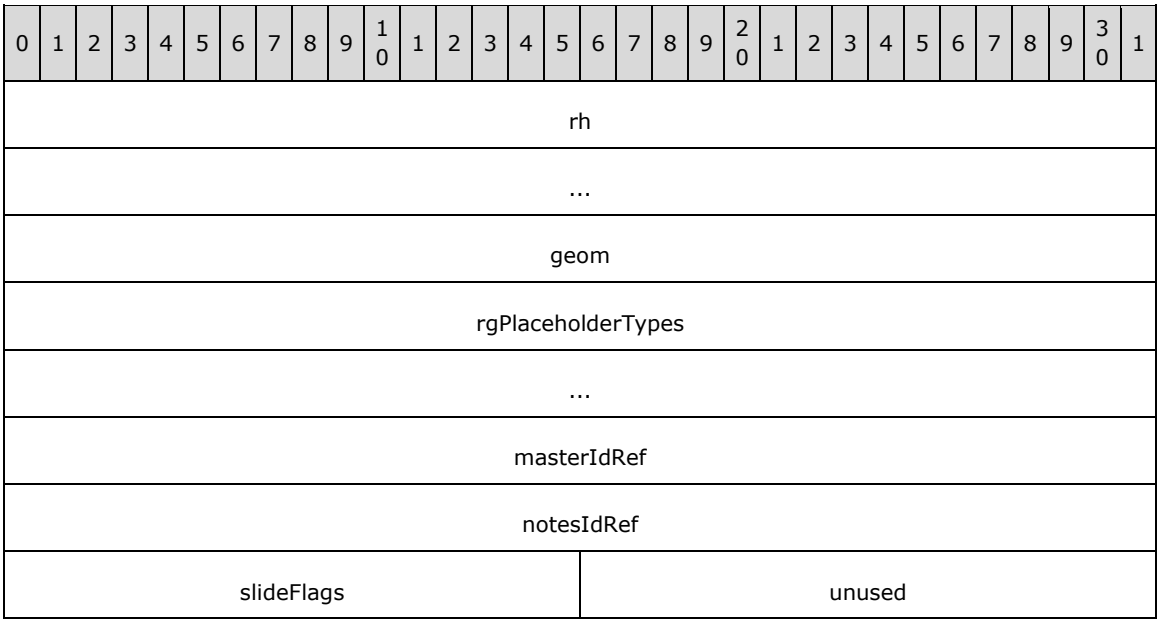
Value	Meaning
RT_RoundTripTheme12Atom	A RoundTripThemeAtom record that specifies round-trip information. It SHOULD<77> be ignored and SHOULD<78> be preserved.
RT_RoundTripColorMapping12Atom	A RoundTripColorMappingAtom record that specifies round-trip information. It SHOULD<79> be ignored and SHOULD<80> be preserved.

2.5.10 SlideAtom

Referenced by: MainMasterContainer, SlideContainer

An atom record that specifies information about a slide.

Let the *corresponding slide* be specified by the **SlideContainer** record (section 2.5.1) or **MainMasterContainer** record (section 2.5.3) that contains this **SlideAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x2.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_SlideAtom.
rh.recLen	MUST be 0x00000018.

geom (4 bytes): A **SlideLayoutType** enumeration that specifies a hint to the user interface which **slide layout** exists on the *corresponding slide*.

A slide layout specifies the type and number of placeholder shapes on a slide. A placeholder shape is specified as an **OfficeArtSpContainer** ([MS-ODRAW] section 2.2.14) that contains a PlaceholderAtom record with a **pos** field not equal to 0xFFFFFFFF. The **placementId** field of the **PlaceholderAtom** record specifies the placeholder shape type. Additional constraints on the type and number of placeholder shapes are specified in the following table.

Value	Placeholder shapes
SL_TitleSlide	The *corresponding slide* MUST be a presentation slide and MUST contain at most one PlaceholderAtom record with a **placementId** equal to PT_CenterTitle, and at most one PlaceholderAtom record with a **placementId** equal to PT_SubTitle, and MUST NOT contain any other PlaceholderAtom record with a **placementId** unequal to 0xFFFFFFFF.
SL_TitleBody	If the *corresponding slide* is a main master slide it MUST contain one PlaceholderAtom record with a **placementId** equal to PT_MasterTitle, one PlaceholderAtom record with a **placementId** equal to PT_MasterBody, one PlaceholderAtom record with a **placementId** equal to PT_MasterDate, one PlaceholderAtom record with a **placementId** equal to PT_MasterFooter, and one PlaceholderAtom record with a **placementId** equal to PT_MasterSlideNumber, and MUST NOT contain any other PlaceholderAtom record with a **placementId** unequal to 0xFFFFFFFF. If the *corresponding slide* is a presentation slide it MUST contain at most one PlaceholderAtom record with **placementId** equal to PT_Title and at most one PlaceholderAtom record with **placementId** equal to PT_Body, PT_Table, PT_OrgChart, PT_Graph, PT_Object or PT_VerticalBody, and MUST NOT contain any other PlaceholderAtom record with a **placementId** unequal to 0xFFFFFFFF.
SL_MasterTitle	The *corresponding slide* MUST be a title master slide and MUST contain one PlaceholderAtom record with a **placementId** equal to PT_MasterCenterTitle, one PlaceholderAtom record with a **placementId** equal to PT_MasterSubTitle, at most one PlaceholderAtom record with a **placementId** equal to PT_MasterDate, at most one PlaceholderAtom record with a **placementId** equal to PT_MasterFooter and at most one PlaceholderAtom record with a **placementId** equal to PT_MasterSlideNumber, and MUST NOT contain any other PlaceholderAtom record with a **placementId** unequal to 0xFFFFFFFF.
SL_TitleOnly	The *corresponding slide* MUST be a presentation slide and MUST contain at most one PlaceholderAtom record with a **placementId** equal to PT_Title, and MUST NOT contain any other PlaceholderAtom record with a **placementId** unequal 0xFFFFFFFF.
SL_TwoColumns	The *corresponding slide* MUST be a presentation slide and MUST contain at most one PlaceholderAtom record with a **placementId** equal to PT_Title, and any combination of zero, one, or two PlaceholderAtom records with **placementId** fields. The following list shows the possible combinations: ▪ PT_Body and PT_Body ▪ PT_Body and PT_Graph ▪ PT_Graph and PT_Body ▪ PT_Body and PT_ClipArt ▪ PT_ClipArt and PT_Body ▪ PT_Body and PT_Object ▪ PT_Object and PT_Body ▪ PT_Body and PT_Media ▪ PT_Media and PT_Body ▪ PT_ClipArt and PT_VerticalBody ▪ PT_Object and PT_Object ▪ PT_Body

Value	Placeholder shapes
	▪ PT_Graph ▪ PT_ClipArt ▪ PT_Object ▪ PT_Media ▪ PT_VerticalBody It MUST NOT contain any other PlaceholderAtom record with a **placementId** unequal to 0xFFFFFFFF.
SL_TwoRows	The *corresponding slide* MUST be a presentation slide and MUST contain at most one PlaceholderAtom record with a **placementId** equal to PT_Title, at most one PlaceholderAtom record with a **placementId** equal to PT_Body, at most one PlaceholderAtom record with a **placementId** equal to PT_Object, and MUST NOT contain any other PlaceholderAtom record with a **placementId** unequal to 0xFFFFFFFF.
SL_ColumnTwoRows	The *corresponding slide* MUST be a presentation slide and MUST contain at most one PlaceholderAtom record with a **placementId** equal to PT_Title, at most one PlaceholderAtom record with a **placementId** equal to PT_Body or PT_Object, at most two additional PlaceholderAtom records with **placementId** fields equal to PT_Object, and MUST NOT contain any other PlaceholderAtom record with a **placementId** unequal to 0xFFFFFFFF.
SL_TwoRowsColumn	The *corresponding slide* MUST be a presentation slide and MUST contain at most one PlaceholderAtom record with a **placementId** equal to PT_Title, at most two PlaceholderAtom records with **placementId** fields equal to PT_Object, at most one additional PlaceholderAtom record with a **placementId** equal to PT_Body or PT_Object, and MUST NOT contain any other PlaceholderAtom record with a **placementId** unequal to 0xFFFFFFFF.
SL_TwoColumnsRow	The *corresponding slide* MUST be a presentation slide and MUST contain at most one PlaceholderAtom record with a **placementId** equal to PT_Title, at most two PlaceholderAtom records with **placementId** fields equal to PT_Object, at most one PlaceholderAtom record with a **placementId** equal to PT_Body, and MUST NOT contain any other PlaceholderAtom record with a **placementId** unequal to 0xFFFFFFFF.
SL_FourObjects	The *corresponding slide* MUST be a presentation slide and MUST contain at most one PlaceholderAtom record with a **placementId** equal to PT_Title, at most four PlaceholderAtom records with **placementId** fields equal to PT_Object, and MUST NOT contain any other PlaceholderAtom record with a **placementId** unequal to 0xFFFFFFFF.
SL_BigObject	The *corresponding slide* MUST be a presentation slide and MUST contain at most one PlaceholderAtom record with a **placementId** equal to PT_Object, and MUST NOT contain any other PlaceholderAtom record with a **placementId** unequal to 0xFFFFFFFF.
SL_Blank	The *corresponding slide* MUST be a presentation slide. There are five layouts supported with this value: Layout 1: The *corresponding slide* MUST NOT contain any PlaceholderAtom records with a **placementId** unequal to 0xFFFFFFFF. Layout 2: The *corresponding slide* MUST contain at most one PlaceholderAtom record with a **placementId** equal to PT_Title, at most one PlaceholderAtom record with a **placementId** equal to PT_Body, and MUST NOT contain any other PlaceholderAtom record with a **placementId** unequal to 0xFFFFFFFF. Layout 3: The *corresponding slide* MUST contain at most one PlaceholderAtom record with a **placementId** equal to PT_Title, at most two PlaceholderAtom records with a **placementId** equal to PT_Body, at most one PlaceholderAtom

Value	Placeholder shapes
	record with a **placementId** equal to PT_Object, and MUST NOT contain any other PlaceholderAtom record with a **placementId** unequal to 0xFFFFFFFF. Layout 4: The *corresponding slide* MUST contain at most one PlaceholderAtom record with a **placementId** equal to PT_Title, at most one PlaceholderAtom record with a **placementId** equal to PT_Object, at most one PlaceholderAtom record with a **placementId** equal to PT_Body, and MUST NOT contain any other PlaceholderAtom record with a **placementId** unequal to 0xFFFFFFFF. Layout 5: The *corresponding slide* MUST contain at most one PlaceholderAtom record with a **placementId** equal to PT_Title, at most one PlaceholderAtom record with a **placementId** equal to PT_Picture, at most one PlaceholderAtom record with a **placementId** equal to PT_Body, and MUST NOT contain any other PlaceholderAtom record with a **placementId** unequal to 0xFFFFFFFF.
SL_VerticalTitleBody	The *corresponding slide* MUST be a presentation slide and MUST contain at most one PlaceholderAtom record with **placementId** equal to PT_VerticalTitle, at most one PlaceholderAtom record with **placementId** equal to PT_VerticalBody, and MUST NOT contain any other PlaceholderAtom record with a **placementId** unequal to 0xFFFFFFFF.
SL_VerticalTwoRows	The *corresponding slide* MUST be a presentation slide and MUST contain at most one PlaceholderAtom record with **placementId** equal to PT_VerticalTitle, at most one PlaceholderAtom record with **placementId** equal to PT_VerticalBody, at most one PlaceholderAtom record with **placementId** equal to PT_Graph, and MUST NOT contain any other PlaceholderAtom record with a **placementId** unequal to 0xFFFFFFFF.

rgPlaceholderTypes (8 bytes): An array of **PlaceholderEnum** (section 2.13.21) enumeration values that specifies a hint to the user interface which placeholder shapes exist on the *corresponding slide*. The count of items in the array MUST be 8. The sequence of array items MUST be a valid **PlaceholderList** as specified by the ABNF (specified in [RFC5234]) grammars in the following table.

Value of geom.	Value of rgPlaceholderTypes

SL_TitleSlide	`PlaceholderList = ( Variant1 / Variant2 )6PT None`
	`Variant1 = PT_CenterTitle PT_SubTitle`
	`Variant2 = PT_Title PT_Body`
	
	The Variant2 rule SHOULD NOT be used.

SL_TitleBody	`PlaceholderList = MasterVariant / SlideVariant`
	`MasterVariant = PT_MasterTitle PT_MasterBody PT_MasterDate PT MasterFooter PT MasterSlideNumber 3PT None`
	`SlideVariant = PT Title ( PT Body / PT Table / PT OrgChart / PT_Graph / PT_Object / PT_VerticalBody )6PT_None`

SL_MasterTitle	`PlaceholderList = PT_MasterCenterTitle PT_MasterSubTitle ( Variant1 / Variant2 )`
	`Variant1 = PT MasterDate PT MasterFooter PT MasterSlideNumber 3PT None`
	`Variant2 = 6PT_None`

Value of geom.	Value of rgPlaceholderTypes
	The Variant2 rule SHOULD NOT be used.
SL_TitleOnly	`PlaceholderList = PT_Title 7PT_None`

SL_TwoColumns	`PlaceholderList = PT_Title ( BodyBody / BodyGraph / GraphBody / BodyClipart / ClipartBody / BodyObject / ObjectBody / BodyMedia / MediaBody / ClipartVBody / ObjectObject ) 5PT None`
	`BodyBody = PT_Body PT_Body`
	`BodyGraph = PT_Body PT_Graph`
	`GraphBody = PT_Graph PT_Body`
	`BodyClipart = PT Body PT ClipArt`
	`ClipartBody = PT_ClipArt PT_Body`
	`BodyObject = PT Body PT Object`
	`ObjectBody = PT_Object PT_Body`
	`BodyMedia = PT_Body PT_Media`
	`MediaBody = PT_Media PT_Body`
	`ClipartVBody = PT ClipArt PT VerticalBody`
	`ObjectObject = PT_Object PT_Object`

SL_TwoRows	`PlaceholderList = PT_Title ( BodyObject / ObjectBody ) 5PT_None`
	`BodyObject = PT_Body PT_Object`
	`ObjectBody = PT_Object PT_Body`

SL_ColumnTwoRows	`PlaceholderList = PT Title ( PT Body / PT Object ) 2PT Object 4PT_None`
SL_TwoRowsColumn	`PlaceholderList = PT_Title 2PT_Object ( PT_Body / PT_Object ) 4PT_None`
SL_TwoColumnsRow	`PlaceholderList = PT_Title 2PT_Object PT_Body 4PT_None`
SL_FourObjects	`PlaceholderList = PT_Title 4PT_Object 3PT_None`
SL_BigObject	`PlaceholderList = PT_Object 7PT_None`

SL_Blank	`PlaceholderList = AllBlank / BlankVariants`
	`AllBlank = 8PT None`
	`BlankVariants = PT_Title ( Variant1 / Variant2 / Variant3 / Variant4 ) 3PT None`
	`Variant1 = PT_Body 3PT_None`
	`Variant2 = PT_Body PT_Object PT_Body PT_Object`
	`Variant3 = PT_Object PT_Body 2PT_None`

Value of geom.	Value of rgPlaceholderTypes
	Variant4 = PT_Picture PT_Body 2PT_None The **BlankVariants**, **Variant1**, **Variant2**, **Variant3,** and **Variant4** rules SHOULD NOT be used.
SL_VerticalTitleBody	PlaceholderList = PT_VerticalTitle PT_VerticalBody 6PT_None
SL_VerticalTwoRows	PlaceholderList = PT VerticalTitle PT VerticalBody PT Graph 5PT_None

masterIdRef (4 bytes): A MasterIdRef that specifies the identifier for the main master slide or title master slide that the *corresponding slide* follows. The value 0x00000000 specifies that the *corresponding slide* does not follow a main master slide or a title master slide. It MUST NOT be 0x00000000 if the record that contains this **SlideAtom** record is a **SlideContainer** section. It MUST be 0x00000000 if the record that contains this **SlideAtom** record is a **MainMasterContainer** record.

notesIdRef (4 bytes): A NotesIdRef that specifies the identifier for the notes slide of the *corresponding slide*. The value 0x00000000 specifies that no notes slide exists. It MUST be 0x00000000 if the record that contains this **SlideAtom** record is a **MainMasterContainer** record.

slideFlags (2 bytes): A SlideFlags structure that specifies which content the *corresponding slide* follows from its main master slide.

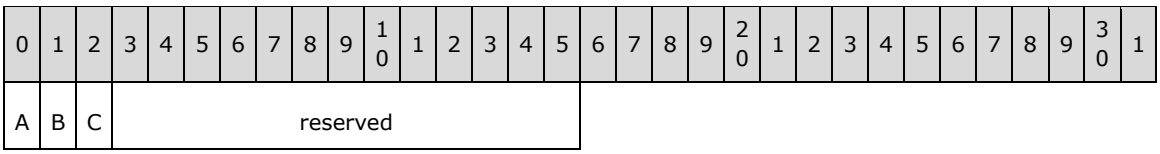
unused (2 bytes): Undefined and MUST be ignored.

2.5.11 SlideFlags

Referenced by: NotesAtom, SlideAtom

A structure that specifies information about a presentation slide and its relationship with its main master slide or title master slide; or about a notes slide and its relationship with its notes master slide.

Let the *corresponding slide* be specified by the **SlideContainer** record (section 2.5.1), **MainMasterContainer** record (section 2.5.3), or **NotesContainer** record (section 2.5.6) that contains this SlideFlags structure.



A - fMasterObjects (1 bit): A bit that specifies whether the *corresponding slide* inherits objects from its main master slide, title master slide, or notes master slide.

B - fMasterScheme (1 bit): A bit that specifies whether the *corresponding slide* inherits the color scheme from its main master slide, title master slide, or notes master slide.

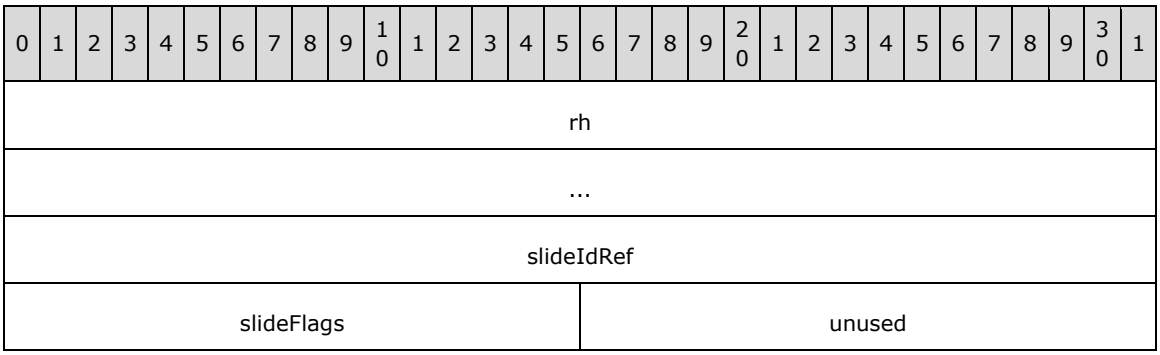
C - fMasterBackground (1 bit): A bit that specifies whether the *corresponding slide* inherits the background from its main master slide, title master slide, or notes master slide.

reserved (13 bits): MUST be zero, and MUST be ignored.

2.5.12 NotesAtom

Referenced by: NotesContainer

An atom record that specifies information about a notes slide or notes master slide.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x1.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_NotesAtom.
rh.recLen	MUST be 0x00000008.

slideIdRef (4 bytes): A **SlideIdRef** (section 2.2.25) that specifies the presentation slide associated with the notes slide. It MUST be 0x00000000 if the **NotesContainer** record (section 2.5.6) that contains this **NotesAtom** record represents the notes master slide. It MUST NOT be 0x00000000 if the **NotesContainer** record that contains this **NotesAtom** record represents a notes slide.

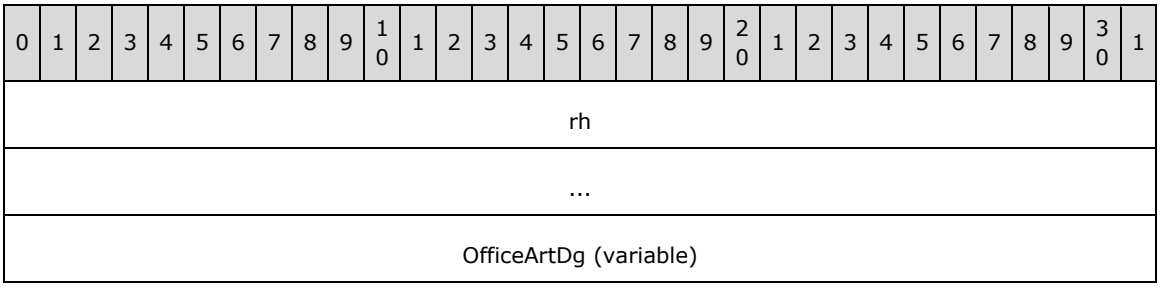
slideFlags (2 bytes): A SlideFlags structure that specifies which content on the notes slide follows content on the notes master slide. It MUST be 0x0000 if **slideIdRef** is 0x00000000.

unused (2 bytes): Undefined and MUST be ignored.

2.5.13 DrawingContainer

Referenced by: HandoutContainer, MainMasterContainer, NotesContainer, SlideContainer

A container record that specifies drawing information for a slide.



...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_Drawing.

OfficeArtDg (variable): An **OfficeArtDgContainer** ([MS-ODRAW] section 2.2.13) that specifies drawing information for a slide.

2.5.14 SlideSchemeColorSchemeAtom

Referenced by: HandoutContainer, MainMasterContainer, NotesContainer, SlideContainer

A container record that specifies the color scheme used by a slide.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
rgSchemeColor (32 bytes)																															
...																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be RT_ColorSchemeAtom.
rh.recLen	MUST be 0x00000020.

rgSchemeColor (32 bytes): An array of ColorStruct structures that specifies a list of colors in the color scheme. The count of items in this array MUST be 8.

2.5.15 SchemeListElementColorSchemeAtom

Referenced by: MainMasterContainer

A container record that specifies a color scheme in a list of available color schemes.

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

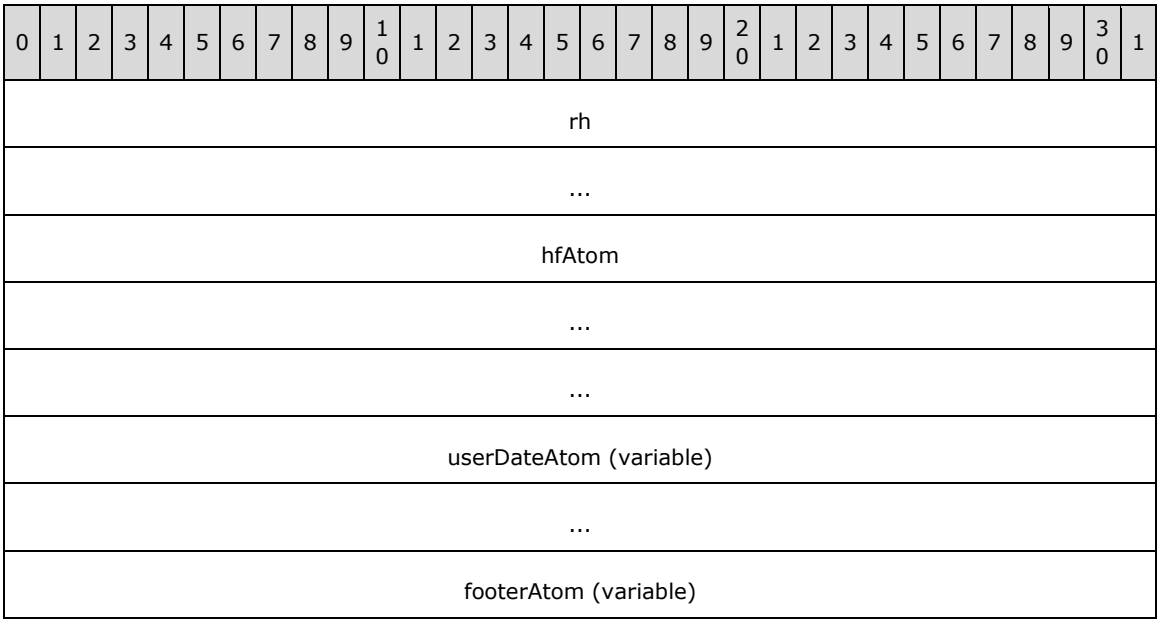
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x006.
rh.recType	MUST be RT_ColorSchemeAtom.
rh.recLen	MUST be 0x00000020.

rgSchemeColor (32 bytes): An array of ColorStruct structures that specifies a list of colors in the color scheme. The count of items in this array MUST be 8.

2.5.16 PerSlideHeadersFootersContainer

Referenced by: MainMasterContainer, SlideContainer

A container record that specifies information about the headers and footers within a slide.



...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_HeadersFooters.

hfAtom (12 bytes): A HeadersFootersAtom record that specifies the options for displaying headers and footers.

userDateAtom (variable): An optional UserDateAtom record that specifies the custom date to be used in the date field.

footerAtom (variable): An optional FooterAtom record that specifies the text that is used in the footer.

2.5.17 SlideNameAtom

Referenced by: HandoutContainer, MainMasterContainer, NotesContainer, SlideContainer

An atom record that specifies the name of a slide.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
slideName (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

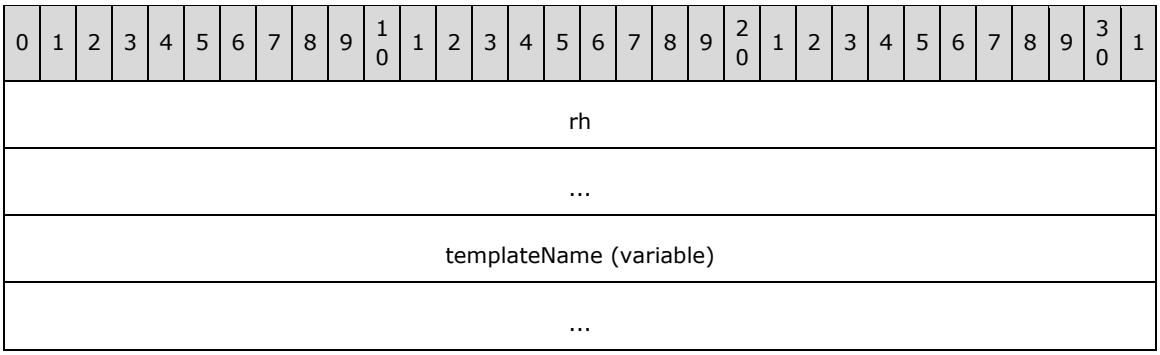
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x003.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number.

slideName (variable): A UnicodeString that specifies the name of a slide. The length, in bytes, of the field is specified by **rh.recLen**.

2.5.18 TemplateNameAtom

Referenced by: MainMasterContainer

An atom record that specifies the name of a main master slide design.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

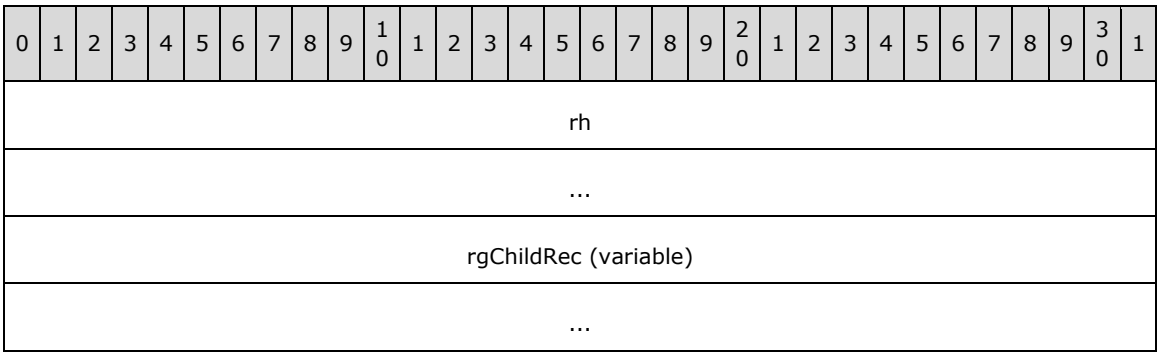
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x002.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 4168.

templateName (variable): A UnicodeString that specifies the name of a main master slide design. The length, in bytes, of the field is specified by **rh.recLen**.

2.5.19 SlideProgTagsContainer

Referenced by: HandoutContainer, MainMasterContainer, NotesContainer, SlideContainer

A container record that specifies programmable tags with additional slide data.



rh (8 bytes): rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.

Field	Meaning
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_ProgTags** (section 2.13.24).

rgChildRec (variable): An array of SlideProgTagsSubContainerOrAtom records that specifies additional slide data. The size, in bytes, of the array is specified by **rh.recLen**. The array MUST NOT contain more than one of each of the following records: PP2.5.23SlideBinaryTagExtension, PP2.5.24SlideBinaryTagExtension, or PP2.5.34SlideBinaryTagExtension.

2.5.20 SlideProgTagsSubContainerOrAtom

Referenced by: SlideProgTagsContainer

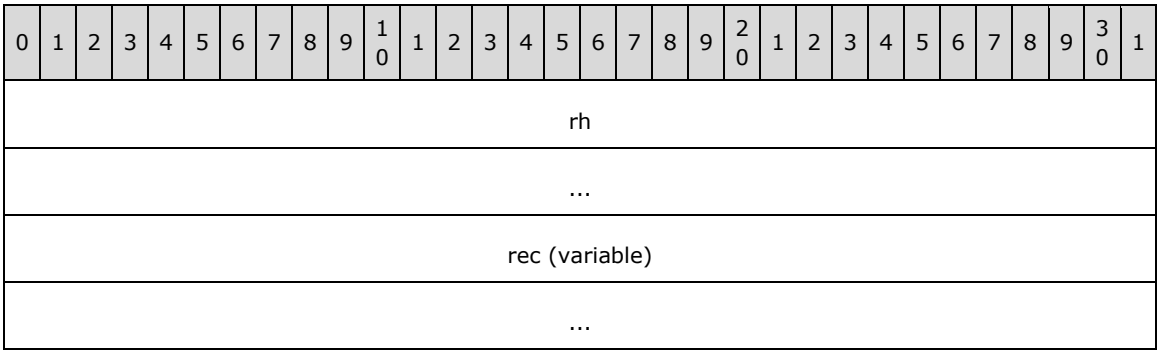
A variable type record whose type and meaning are dictated by the value of **rh.recType**, as specified in the following table.

Value	Meaning
RT_ProgStringTag	A ProgStringTagContainer record that specifies additional slide data.
RT_ProgBinaryTag	A SlideProgBinaryTagContainer record that specifies additional slide data.

2.5.21 SlideProgBinaryTagContainer

Referenced by: SlideProgTagsSubContainerOrAtom

A container record that specifies programmable tags with additional binary slide data.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_ProgBinaryTag.

rec (variable): A SlideProgBinaryTagSubContainerOrAtom record that specifies additional slide data.

2.5.22 SlideProgBinaryTagSubContainerOrAtom

Referenced by: SlideProgBinaryTagContainer

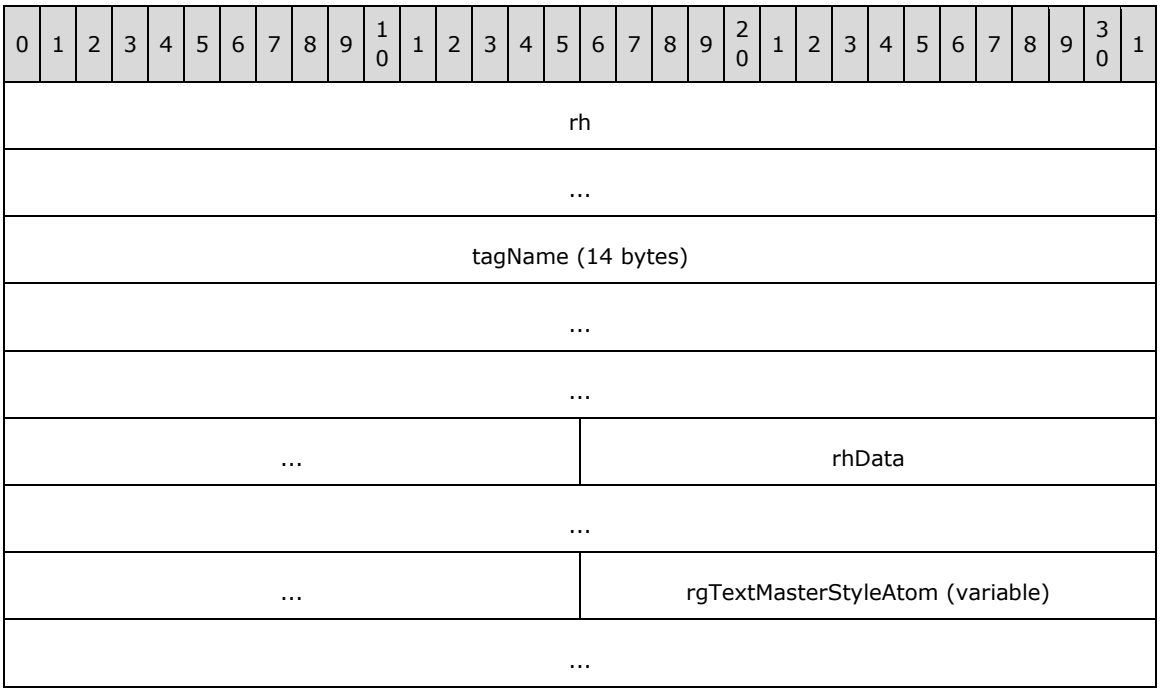
A variable type record whose type and meaning are dictated by the value of **tagNameAtom.tagName** for UnknownBinaryTag or by the value of **tagName** for PP2.5.23SlideBinaryTagExtension, PP2.5.24SlideBinaryTagExtension, PP2.5.34SlideBinaryTagExtension, as specified in the following table.

Value	Meaning
"___PPT9"	A PP2.5.23SlideBinaryTagExtension record pair that specifies additional slide data. It MAY<81> be ignored and MUST be preserved.
"___PPT10"	A PP2.5.24SlideBinaryTagExtension record pair that specifies additional slide data. It MAY<82> be ignored and MUST be preserved.
"___PPT12"	A PP2.5.34SlideBinaryTagExtension record pair that specifies additional slide data. It MAY<83> be ignored and MUST be preserved.
Any other value	An UnknownBinaryTag record pair that specifies additional slide data. It MUST be ignored and MUST be preserved.

2.5.23 PP9SlideBinaryTagExtension

Referenced by: SlideProgBinaryTagSubContainerOrAtom

A pair of atom records that specifies a programmable tag with additional slide data.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the first record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.

Field	Meaning
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be 0x0000000E.

tagName (14 bytes): A **PrintableUnicodeString** string (section 2.2.23) that specifies the programmable tag name. It MUST be "___PPT9".

rhData (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the second record. Sub-fields are further specified in the following table.

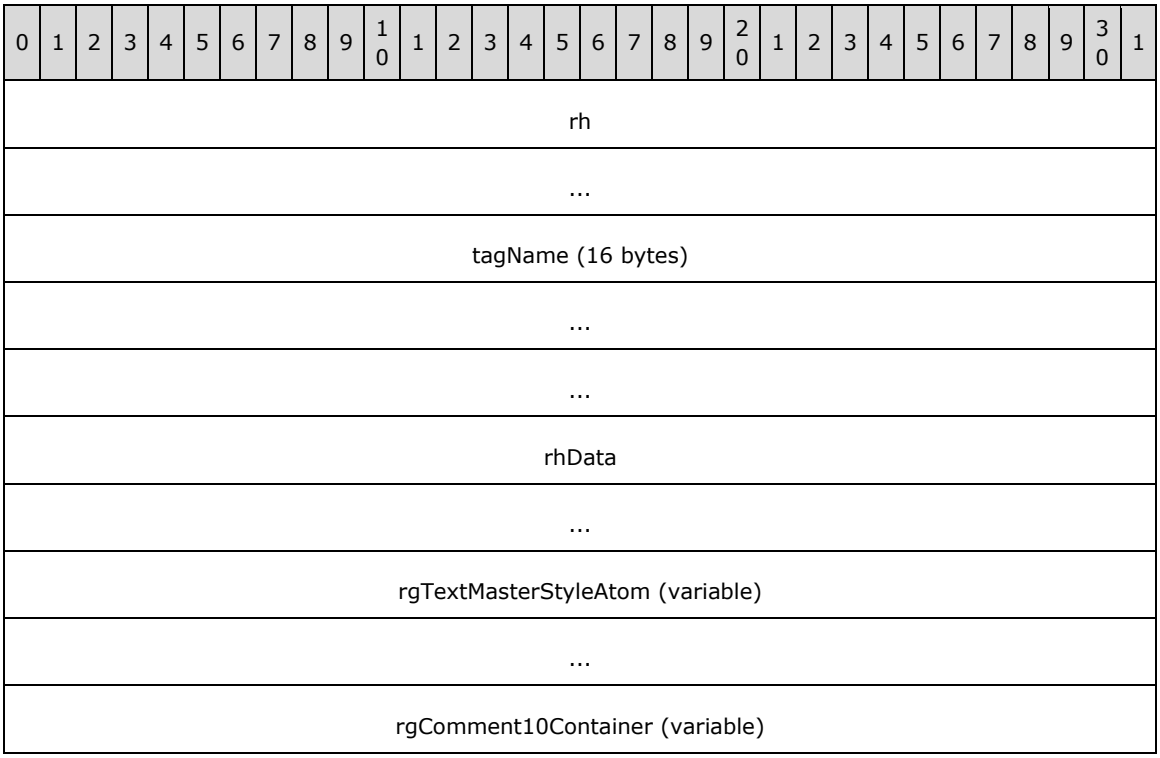
Field	Meaning
rhData.recVer	MUST be 0x0.
rhData.recInstance	MUST be 0x000.
rhData.recType	MUST be RT_BinaryTagDataBlob.

rgTextMasterStyleAtom (variable): An array of TextMasterStyle2.9.37Atom records that specifies additional character-level and paragraph-level formatting of master slides. The size, in bytes, of the array is specified by **rhData.recLen**.

2.5.24 PP10SlideBinaryTagExtension

Referenced by: SlideProgBinaryTagSubContainerOrAtom

A pair of atom records that specifies a programmable tag with additional slide data.



...
linkedSlideAtom (16 bytes, optional)
...
...
rgLinkedShape10Atom (variable)
...
slideFlagsAtom (optional)
...
...
slideTimeAtom (16 bytes, optional)
...
...
hashCodeAtom (optional)
...
...
extTimeNodeContainer (variable)
...
buildListContainer (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the first record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be 0x00000010.

tagName (16 bytes): A **PrintableUnicodeString** string (section 2.2.23) that specifies the programmable tag name. It MUST be "___PPT10".

rhData (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the second record. Sub-fields are further specified in the following table.

Field	Meaning
rhData.recVer	MUST be 0x0.
rhData.recInstance	MUST be 0x000.
rhData.recType	MUST be RT_BinaryTagDataBlob.

rgTextMasterStyleAtom (variable): An array of TextMasterStyle2.9.39Atom records that specifies additional character-level and paragraph-level formatting of master slides. The array continues while **rh.recType** of the TextMasterStyle2.9.39Atom item is equal to RT_TextMasterStyle10Atom.

rgComment10Container (variable): An array of Comment2.5.25Container records that specifies presentation comments. The array continues while **rh.recType** of the Comment2.5.25Container item is equal to RT_Comment10.

linkedSlideAtom (16 bytes): An optional LinkedSlide2.5.32Atom record that specifies a link to a slide used to display the changes to the slide made by a reviewer. It SHOULD<84> be ignored and SHOULD<85> be omitted.

rgLinkedShape10Atom (variable): An optional array of LinkedShape2.5.33Atom records. The count of items in the array is specified by **linkedSlideAtom.cLinkedShapes**. The array specifies links to shapes used to display the changes of the slide made by a reviewer. It SHOULD<86> be ignored and SHOULD<87> be omitted.

slideFlagsAtom (12 bytes): An optional SlideFlags2.5.30Atom record that specifies slide-level flags.

slideTimeAtom (16 bytes): An optional SlideTime2.5.31Atom record that specifies the slide creation timestamp.

hashCodeAtom (12 bytes): An optional HashCode2.8.3Atom record that specifies a hash code for the animations on the slide.

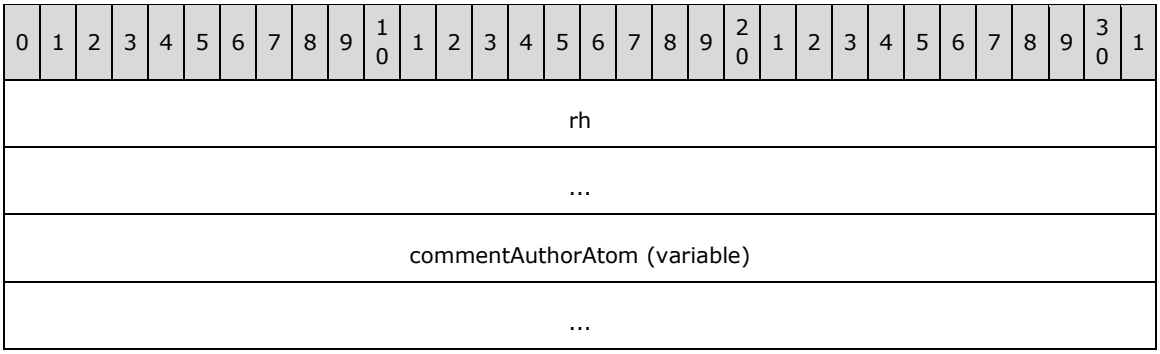
extTimeNodeContainer (variable): An optional **ExtTimeNodeContainer** record (section 2.8.15) that specifies slide animation timing data.

buildListContainer (variable): An optional BuildListContainer record that specifies slide animation **build** data.

2.5.25 Comment10Container

Referenced by: PP10SlideBinaryTagExtension

A container record that specifies a presentation comment.



commentTextAtom (variable)
...
commentAuthorInitialsAtom (variable)
...
commentAtom (36 bytes)
...
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_Comment10.

commentAuthorAtom (variable): An optional Comment2.5.26AuthorAtom record that specifies the name of the author of the presentation comment.

commentTextAtom (variable): An optional Comment2.5.27TextAtom record that specifies the text of the presentation comment.

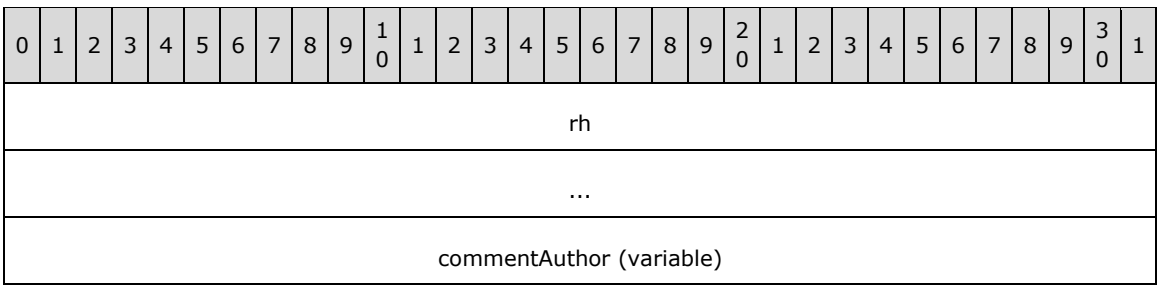
commentAuthorInitialsAtom (variable): An optional Comment2.5.28AuthorInitialAtom record that specifies the initials of the author of the presentation comment.

commentAtom (36 bytes): A Comment2.5.29Atom record that specifies the settings for displaying the presentation comment.

2.5.26 Comment10AuthorAtom

Referenced by: Comment10Container

An atom record that specifies the name of the author of the presentation comment. The presentation comment is specified by the Comment2.5.25Container record that contains this **Comment10AuthorAtom** record.



...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 104.

commentAuthor (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the name of the author. The length, in bytes, of the field is specified by **rh.recLen**.

2.5.27 Comment10TextAtom

Referenced by: Comment10Container

An atom record that specifies the text of the presentation comment. The presentation comment is specified by the Comment2.5.25Container record that contains this **Comment10TextAtom** record.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
commentText (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

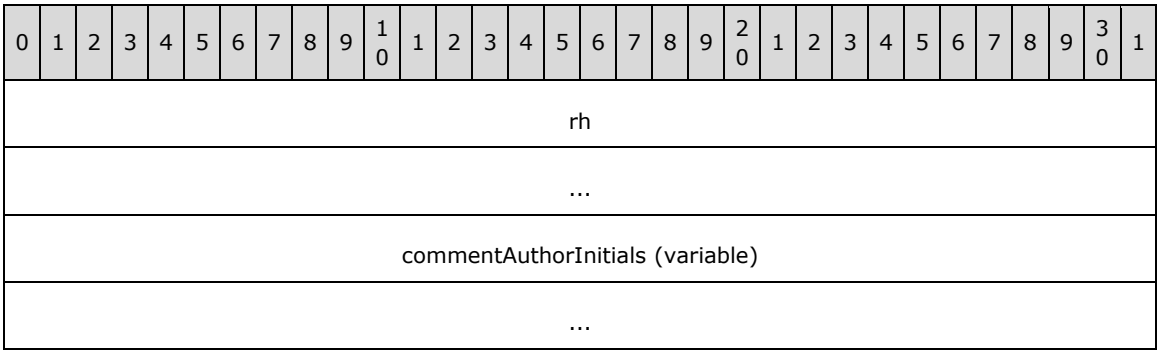
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be an **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 64000.

commentText (variable): A TabCrLfPrintableUnicodeString that specifies the text for the presentation comment. The length, in bytes, of the field is specified by **rh.recLen**.

2.5.28 Comment10AuthorInitialAtom

Referenced by: Comment10Container

An atom record that specifies the initials of the author of the presentation comment. The presentation comment is specified by the Comment2.5.25Container record that contains this **Comment10AuthorInitialAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

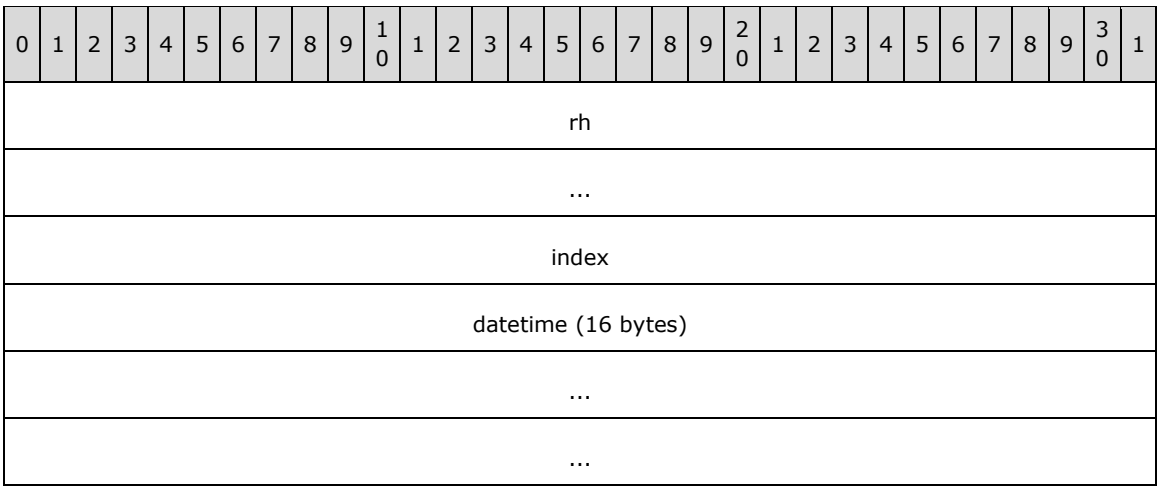
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x002.
rh.recType	MUST be an **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be less than or equal to 104.

commentAuthorInitials (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the initials of the author of the presentation comment. The length, in bytes, of the field is specified by **rh.recLen**.

2.5.29 Comment10Atom

Referenced by: Comment10Container

An atom record that specifies the settings for displaying a presentation comment. The presentation comment is specified by the Comment2.5.25Container record that contains this **Comment10Atom** record.



anchor
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_Comment10Atom.
rh.recLen	MUST be 0x0000001C.

index (4 bytes): A signed integer that specifies the index of the presentation comment. The index is part of the presentation comment label. It MUST be greater than or equal to zero.

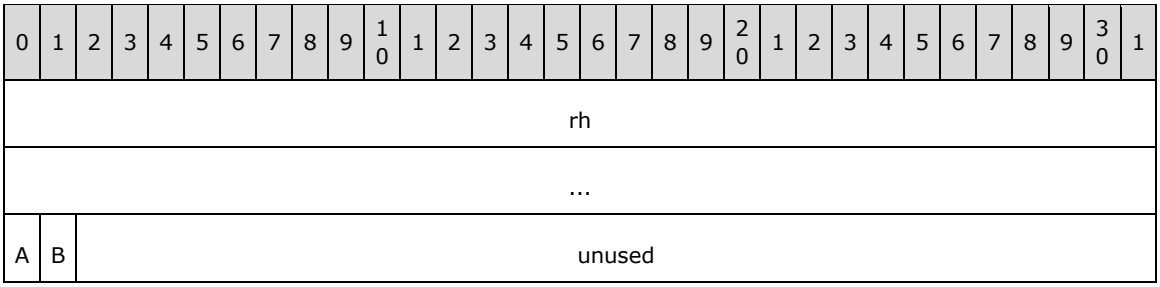
datetime (16 bytes): A DateTimeStruct structure that specifies the creation time of the presentation comment.

anchor (8 bytes): A **PointStruct** structure (section 2.12.5) that specifies the location of the presentation comment label in master units, relative to the top-left corner of the slide.

2.5.30 SlideFlags10Atom

Referenced by: PP10SlideBinaryTagExtension

An atom record that specifies slide-level flags.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_SlideFlags10Atom.
rh.recLen	MUST be 0x00000004.

A - fPreserveMaster (1 bit): A bit that specifies whether to preserve the main master slide or title master slide when there is no slide that follows it. It MUST be ignored if the slide is not a main master slide or title master slide.

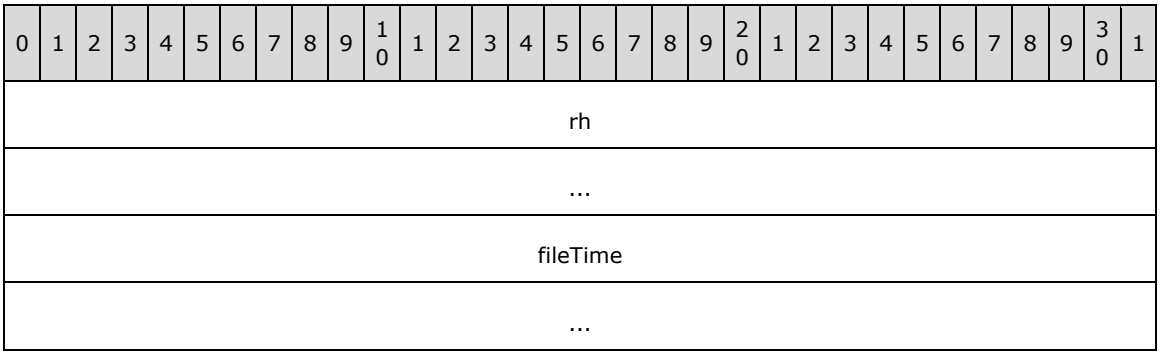
B - fOverrideMasterAnimation (1 bit): A bit that specifies whether the slide does not follow the animations on the main master slide or title master slide.

unused (30 bits): Undefined and MUST be ignored.

2.5.31 SlideTime10Atom

Referenced by: PP10SlideBinaryTagExtension

An atom record that specifies the slide creation time stamp.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_SlideTime10Atom.
rh.recLen	MUST be 0x00000008.

fileTime (8 bytes): A **FILETIME** structure, as specified in [MS-DTYP] section 2.3.3, that specifies the time of slide creation.

2.5.32 LinkedSlide10Atom

Referenced by: PP10SlideBinaryTagExtension

An atom record that specifies a reference to a presentation slide and a count of LinkedShape2.5.33Atom records.

Let the *corresponding main document* be specified by the document that contains both the *corresponding linked document* and the *corresponding reviewer document*.

Let the *corresponding linked document* be specified as follows:

- Let the *corresponding main master slide* be specified by the **MainMasterContainer** record (section 2.5.3) that is specified by the first **MasterPersistAtom** record (section 2.4.14.2) in the **MasterListWithTextContainer** record (section 2.4.14.1) in the *corresponding main document*.
- Let the *corresponding shape* be specified by the **OfficeArtSpContainer** record ([MS-ODRAW] section 2.2.14) such that the **wzName_complex** property ([MS-ODRAW] section 2.3.4.2) matches the string "Linked". The *corresponding shape* is contained by the **drawing** field of the *corresponding main master slide*.

- Let the *corresponding OLE object* be specified by the **ExOleEmbedContainer** record (section 2.10.27) whose **exOleObjAtom.exObjId** field matches the **exObjIdRef** field of the ExObjRefAtom record that is contained within the *corresponding shape*. Let the *corresponding linked document* be the *corresponding OLE object* so specified.

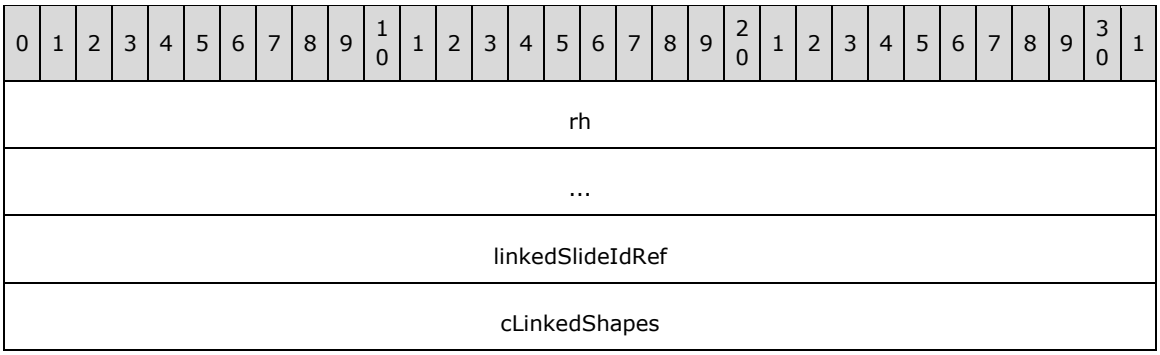
Let the *corresponding reviewer document* be specified as follows:

- Let the *corresponding main master slide* be specified by the **MainMasterContainer** record that is specified by the first **MasterPersistAtom** record (section 2.4.14.2) in the **MasterListWithTextContainer** record (section 2.4.14.1) in the *corresponding main document*.
- Let the *corresponding shape* be specified by the **OfficeArtSpContainer** record ([MS-ODRAW] section 2.2.14) such that the **wzName_complex** property ([MS-ODRAW] section 2.3.4.2) matches the string "Reviewer". The *corresponding shape* is contained by the **drawing** field of the *corresponding main master slide*.
- Let the *corresponding OLE object* be specified by the **ExOleEmbedContainer** record whose **exOleObjAtom.exObjId** field matches the **exObjIdRef** field of the ExObjRefAtom record that is contained within the *corresponding shape*. Let the *corresponding reviewer document* be the *corresponding OLE object* so specified.

Let the *corresponding base document* be specified as follows:

- Let the *corresponding main master slide* be specified by the **MainMasterContainer** record that is specified by the first **MasterPersistAtom** record (section 2.4.14.2) in the **MasterListWithTextContainer** record (section 2.4.14.1) in the *corresponding reviewer document*.
- Let the *corresponding shape* be specified by the **OfficeArtSpContainer** record ([MS-ODRAW] section 2.2.14) such that the **wzName_complex** property ([MS-ODRAW] section 2.3.4.2) matches the string "Base". The *corresponding shape* is contained by the **drawing** field of the *corresponding main master slide*.
- Let the *corresponding OLE object* be specified by the **ExOleEmbedContainer** record whose **exOleObjAtom.exObjId** field matches the **exObjIdRef** field of the ExObjRefAtom record that is contained within the *corresponding shape*. Let the *corresponding base document* be the *corresponding OLE object* so specified.

If this **LinkedSlide10Atom** record is contained within the *corresponding main document*, let the *associated document* be specified by the *corresponding linked document*; or if this **LinkedSlide10Atom** record is contained within the *corresponding reviewer document*, let the *associated document* be specified by the *corresponding main document*; or if this **LinkedSlide10Atom** record is contained within the *corresponding base document*, let the associated document be specified by the *corresponding linked document*.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_LinkedSlide10Atom.
rh.recLen	MUST be 0x00000008.

linkedSlideIdRef (4 bytes): A **SlideIdRef** (section 2.2.25) that specifies a reference to a presentation slide in the *associated document*.

cLinkedShapes (4 bytes): A **signed integer** that specifies the count of LinkedShape2.5.33Atom records in the **rgLinkedShape10Atom** field of the PP2.5.24SlideBinaryTagExtension record that contains this **LinkedSlide10Atom** record.

2.5.33 LinkedShape10Atom

Referenced by: PP10SlideBinaryTagExtension

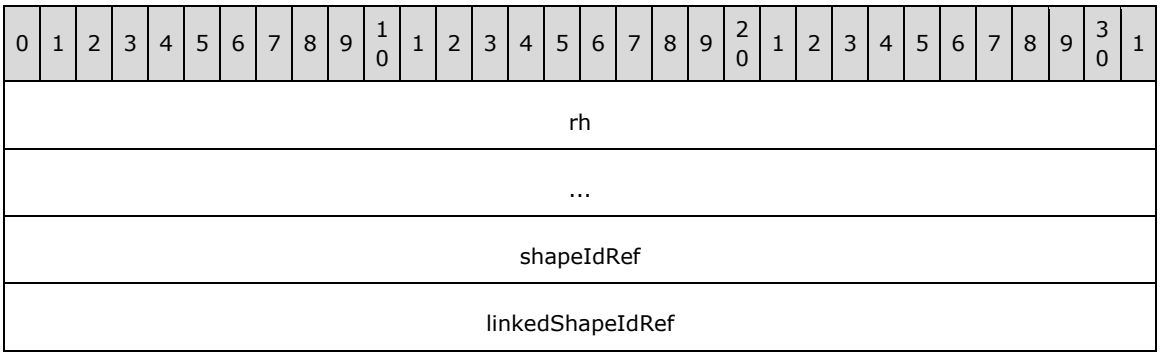
An atom record that specifies a shape in a presentation slide that contains this **LinkedShape10Atom** record and its *associated shape* from the *associated presentation slide* that is contained within the *associated document*.

Let the *corresponding slide* be specified by the **SlideContainer** record (section 2.5.1) that contains this **LinkedShape10Atom** record.

Let the *associated document* be as specified by the LinkedSlide2.5.32Atom record contained within the PP2.5.24SlideBinaryTagExtension record pair that contains this **LinkedShape10Atom** record.

Let the *associated presentation slide* contained within the *associated document* be specified by the **linkedSlideIdRef** field of the LinkedSlide2.5.32Atom record in the PP2.5.24SlideBinaryTagExtension record pair that contains this **LinkedShape10Atom** record.

Let the *associated shape* contained within the *associated presentation slide* be specified by the **linkedShapeIdRef** field of this **LinedShape10Atom**.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_LinkedShape10Atom.
rh.recLen	MUST be 0x00000008.

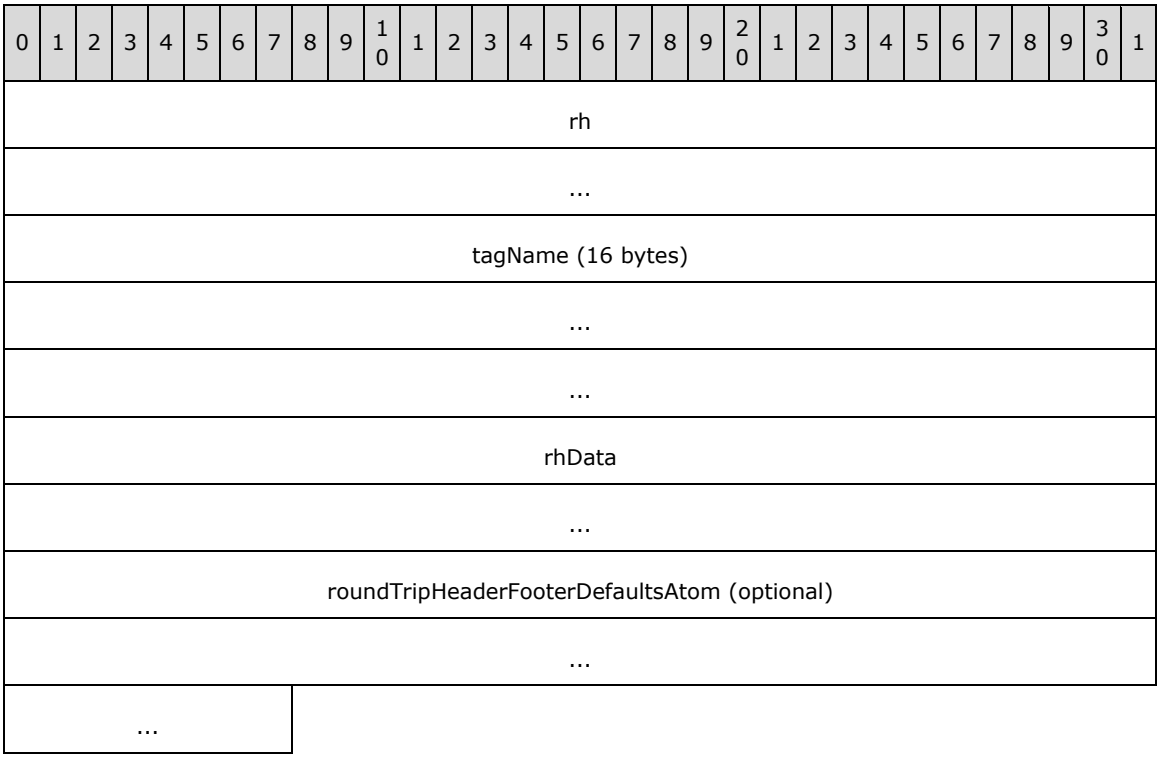
shapeIdRef (4 bytes): An **unsigned integer** that specifies a reference to the **OfficeArtSpContainer** ([MS-ODRAW] section 2.2.14) contained within the *corresponding slide* such that the **shapeProp.spid** field matches the value of this field.

linkedShapeIdRef (4 bytes): An **unsigned integer** that specifies a reference to the **OfficeArtSpContainer** ([MS-ODRAW] section 2.2.14) contained within the *associated presentation slide* such that the **shapeProp.spid** field matches the value of this field.

2.5.34 PP12SlideBinaryTagExtension

Referenced by: SlideProgBinaryTagSubContainerOrAtom

A pair of atom records that specifies a programmable tag with additional slide data.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the first record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be 0x00000010.

tagName (16 bytes): A **PrintableUnicodeString** string (section 2.2.23) that specifies the programmable tag name. It MUST be "___PPT12".

rhData (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the second record. Sub-fields are further specified in the following table.

Field	Meaning
rhData.recVer	MUST be 0x0.
rhData.recInstance	MUST be 0x000.
rhData.recType	MUST be RT_BinaryTagDataBlob.

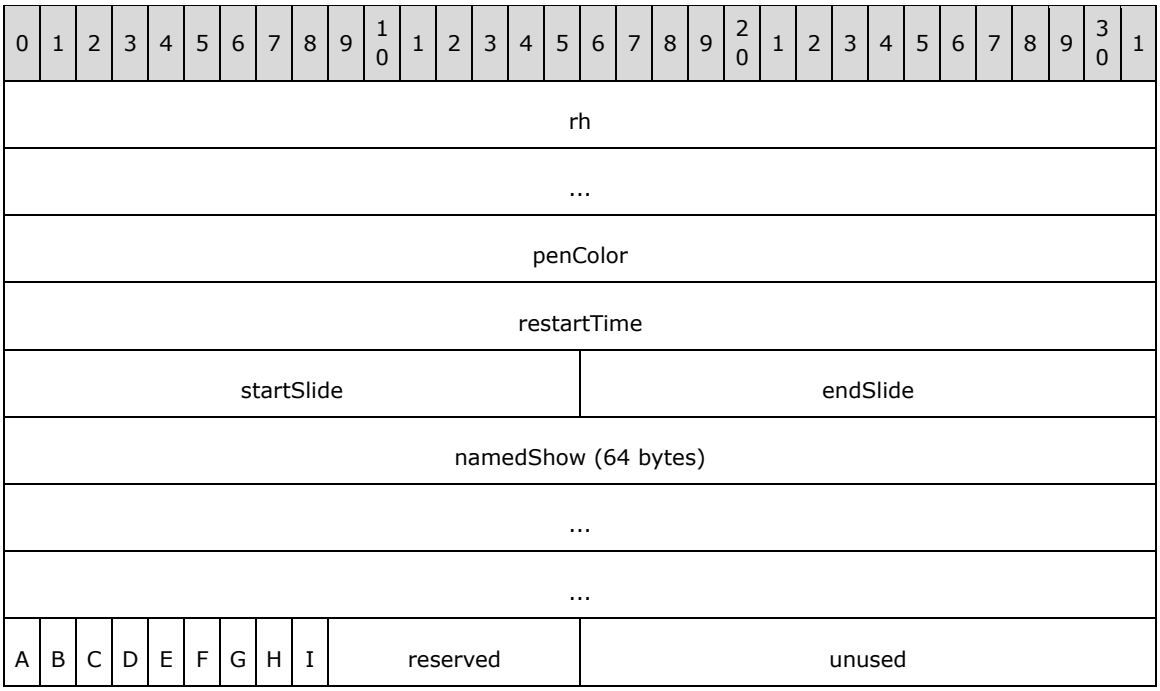
roundTripHeaderFooterDefaultsAtom (9 bytes): An optional RoundTripHeaderFooterDefaults2.11.15Atom record that specifies headers and footers data for a main master slide, title master slide, handout master slide, or notes master slide.

2.6 Slide Show Types

2.6.1 SlideShowDocInfoAtom

Referenced by: DocumentContainer

An atom record that specifies how a slide show is displayed.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x1.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_SlideShowDocInfoAtom.
rh.recLen	MUST be 0x00000050.

penColor (4 bytes): A ColorIndexStruct structure that specifies a color used to annotate presentation slides during a slide show.

restartTime (4 bytes): A **signed integer** that specifies an amount of time, in milliseconds, to wait during a period of inactivity before restarting a slide show in **kiosk mode**.

startSlide (2 bytes): A **signed integer** that specifies the one-based index of the slide with which the slide show starts. It MUST be greater than or equal to 0x0000. If **fUseSlideRange** is **TRUE**, it MUST NOT be set to 0x0000.

endSlide (2 bytes): A **signed integer** that specifies the one-based index of the slide with which the slide show ends. It MUST be greater than or equal to 0x0000. If **fUseSlideRange** is **TRUE**, it MUST NOT be set to 0x0000.

namedShow (64 bytes): A char2.2.4 that specifies the name of a named show to use when running the slide show.

A - fAutoAdvance (1 bit): A bit that specifies whether to automatically advance to the next slide during a slide show based on timing information on the slide.

B - fWillSkipBuilds (1 bit): A bit that specifies whether to display animations during a slide show.

C - fUseSlideRange (1 bit): A bit that specifies whether to display only the slide range specified by the **startSlide** and **endSlide** fields during a slide show.

D - fDocUseNamedShow (1 bit): A bit that specifies whether the slides shown during a slide show are from the named show identified by **namedShow**. It MUST be ignored if **fUseSlideRange** is **TRUE**.

E - fBrowseMode (1 bit): A bit that specifies whether the slide show is presented in a way optimized for browsing. If **fBrowseMode** is **TRUE**, **fKioskMode** MUST be **FALSE**.

F - fKioskMode (1 bit): A bit that specifies whether the slide show is presented in a way optimized to run at a kiosk. If **fKioskMode** is **TRUE**, **fBrowseMode** MUST be **FALSE**.

G - fWillSkipNarration (1 bit): A bit that specifies whether to play slide audio narrations during a slide show.

H - fLoopContinuously (1 bit): A bit that specifies whether to restart the slide show at the beginning after advancing from the last slide.

I - fHideScrollBar (1 bit): A bit that specifies whether to display the navigational scroll bar during a slide show.

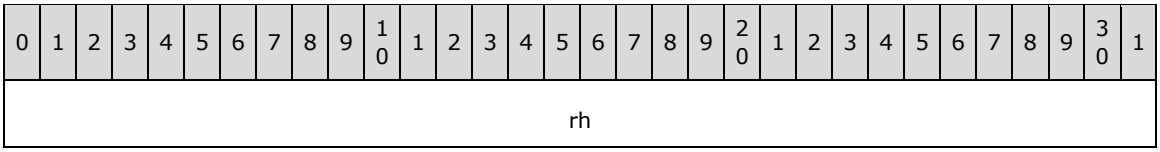
reserved (7 bits): MUST be zero and MUST be ignored.

unused (2 bytes): Undefined and MUST be ignored.

2.6.2 NamedShowsContainer

Referenced by: DocumentContainer

A container record that specifies the named shows.



...
rgNamedShow (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_NamedShows.

rgNamedShow (variable): An array of NamedShowContainer records that specifies the named shows.

2.6.3 NamedShowContainer

Referenced by: NamedShowsContainer

A container record that specifies a named show.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
namedShowNameAtom (variable)																															
...																															
namedShowSlidesAtom (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_NamedShow.

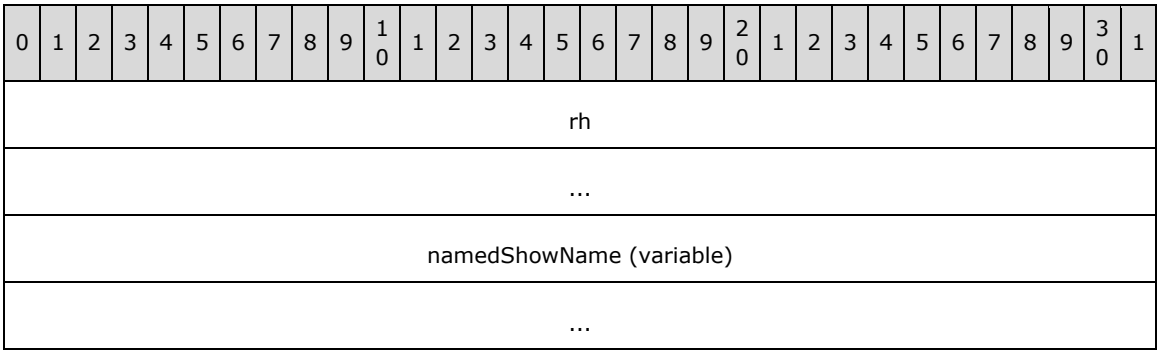
namedShowNameAtom (variable): A NamedShowNameAtom record that specifies the name of this named show.

namedShowSlidesAtom (variable): An optional NamedShowSlidesAtom record that specifies the slides in this named show.

2.6.4 NamedShowNameAtom

Referenced by: NamedShowContainer

An atom record that specifies the name of a named show.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

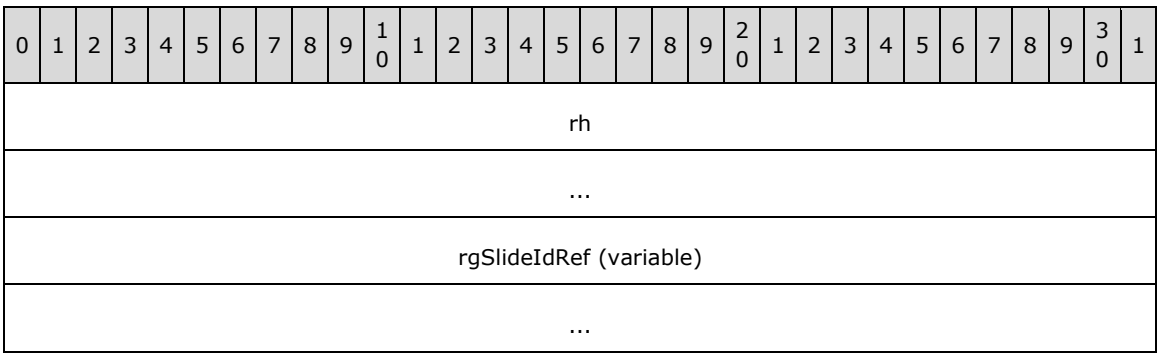
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number.

namedShowName (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the name of a named show that can be used when running the slide show.

2.6.5 NamedShowSlidesAtom

Referenced by: NamedShowContainer

An atom record that specifies the slide identifiers of presentation slides in a named show.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_NamedShowSlidesAtom.

rgSlideIdRef (variable): An array of **SlideIdRef** (section 2.2.25) that specifies the slides that are in this named show. The order of the slides in this array is also the order for the slides in the named show. Any slides referenced here that do not exist in the presentation itself MUST be ignored.

2.6.6 SlideShowSlideInfoAtom

Referenced by: MainMasterContainer, SlideContainer

An atom record that specifies what transition effect to perform during a slide show, and how to advance to the next presentation slide.

Let the *corresponding slide* be specified by the **SlideContainer** record (section 2.5.1) that contains this **SlideShowSlideInfoAtom** record.

<table>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>20</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>30</td><td>1</td></tr>
<tr><td colspan="32">rh</td></tr>
<tr><td colspan="32">...</td></tr>
<tr><td colspan="32">slideTime</td></tr>
<tr><td colspan="32">soundIdRef</td></tr>
<tr><td colspan="8">effectDirection</td><td colspan="8">effectType</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td>G</td><td>H</td><td>I</td><td>J</td><td>K</td><td>L</td><td>M</td><td colspan="5">N</td></tr>
<tr><td colspan="8">speed</td><td colspan="24">unused</td></tr>
</table>

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table:

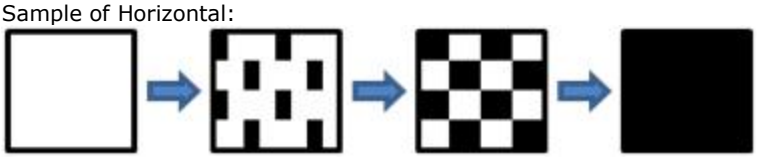
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_SlideShowSlideInfoAtom.
rh.recLen	MUST be 0x00000010.

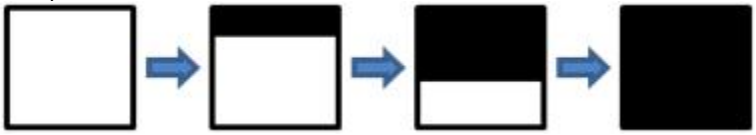
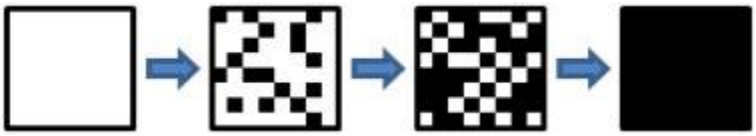
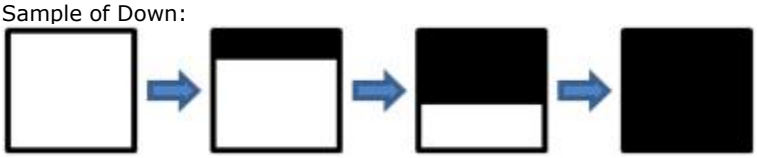
slideTime (4 bytes): A signed integer that specifies an amount of time, in milliseconds, to wait before advancing to the next presentation slide. It MUST be greater than or equal to 0 and less than or equal to 86399000. It MUST be ignored unless **fAutoAdvance** is **TRUE**.

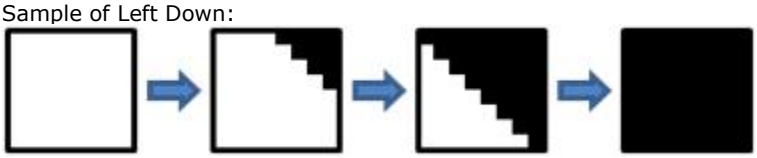
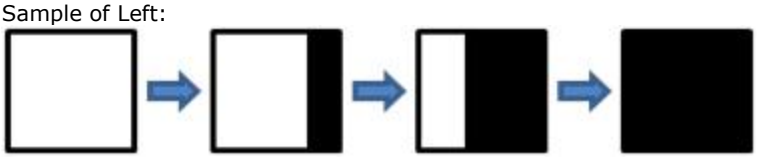
soundIdRef (4 bytes): A SoundIdRef that specifies which sound to play when the transition starts.

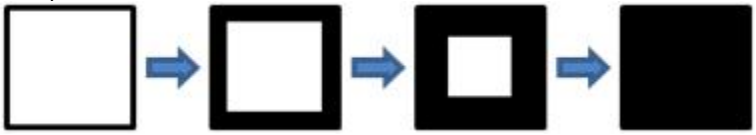
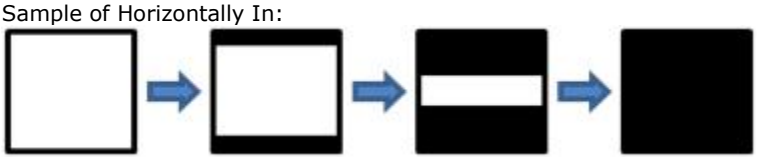
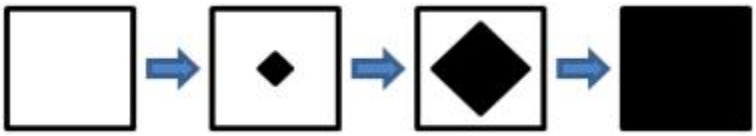
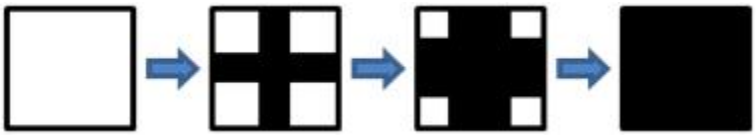
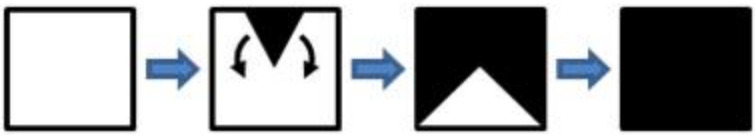
effectDirection (1 byte): A byte that specifies the variant of **effectType**. See the **effectType** field for further restriction and specification of this field.

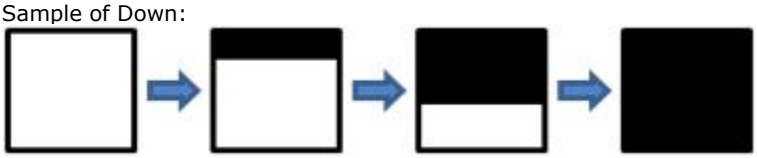
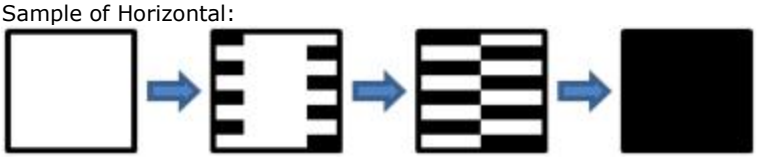
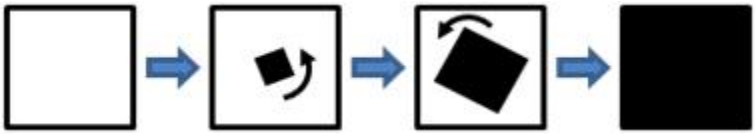
effectType (1 byte): A byte that specifies which transition is used when transitioning to the next presentation slide during a slide show. Any of the following samples are for sample purposes only. Exact rendering of any transition is determined by the rendering application. As such, the same transition can have many variations depending on the implementation.

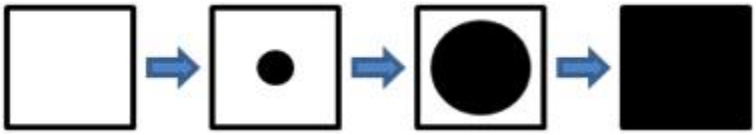
Value	Meaning
0	Cut The following specifies the possible **effectDirection** values and their meanings: ▪ 0x00: The transition is not made through black. (The effect is the same as no transition at all.) ▪ 0x01: The transition is made through black. Sample of through black: 
1	Random **effectDirection** MUST be ignored.
2	Blinds The following specifies the possible **effectDirection** values and their meanings: ▪ 0x00: Vertical ▪ 0x01: Horizontal
3	Checker The following specifies the possible **effectDirection** values and their meanings: ▪ 0x00: Horizontal ▪ 0x01: Vertical Sample of Horizontal: 
4	Cover The following specifies the possible **effectDirection** values and their meanings: ▪ 0x00: Left ▪ 0x01: Up ▪ 0x02: Right ▪ 0x03: Down

Value	Meaning
	▪ 0x04: Left Up ▪ 0x05: Right Up ▪ 0x06: Left Down ▪ 0x07: Right Down Sample of Down: 
5	Dissolve  **effectDirection** MUST be 0x00.
6	Fade  **effectDirection** MUST be 0x00.
7	Uncover The following specifies the possible **effectDirection** values and their meanings: ▪ 0x00: Left ▪ 0x01: Up ▪ 0x02: Right ▪ 0x03: Down ▪ 0x04: Left Up ▪ 0x05: Right Up ▪ 0x06: Left Down ▪ 0x07: Right Down Sample of Down: 

Value	Meaning
8	Random Bars The following specifies the possible **effectDirection** values and their meanings: ▪ 0x00: Horizontal ▪ 0x01: Vertical Sample of Horizontal: 
9	Strips The following specifies the possible **effectDirection** values and their meanings: ▪ 0x04: Left Up ▪ 0x05: Right Up ▪ 0x06: Left Down ▪ 0x07: Right Down Sample of Left Down: 
10	Wipe The following specifies the possible **effectDirection** values and their meanings: ▪ 0x00: Left ▪ 0x01: Up ▪ 0x02: Right ▪ 0x03: Down Sample of Left: 
11	Box In/Out The following specifies the possible **effectDirection** values and their meanings: ▪ 0x00: Out

Value	Meaning
	▪ 0x01: In Sample of In: 
13	Split The following specifies the possible **effectDirection** values and their meanings: ▪ 0x00: Horizontally out ▪ 0x01: Horizontally in ▪ 0x02: Vertically out ▪ 0x03: Vertically in Sample of Horizontally In: 
17	Diamond  **effectDirection** MUST be 0x00.
18	Plus  **effectDirection** MUST be 0x00.
19	Wedge  **effectDirection** MUST be 0x00.
20	Push The following specifies the possible **effectDirection** values and their meanings:

Value	Meaning
	▪ 0x00: Left ▪ 0x01: Up ▪ 0x02: Right ▪ 0x03: Down Sample of Down: 
21	Comb The following specifies the possible **effectDirection** values and their meanings: ▪ 0x00: Horizontal ▪ 0x01: Vertical Sample of Horizontal: 
22	Newsflash  **effectDirection** MUST be 0x00.
23	AlphaFade  **effectDirection** MUST be 0x00.
26	Wheel Sample of 3: 

Value	Meaning
	effectDirection values refer to the number of radial divisions used in the effect. The value MUST be one of 0x01, 0x02, 0x03, 0x04, or 0x08.
27	Circle  **effectDirection** MUST be 0x00.
255	Undefined and MUST be ignored.

A - fManualAdvance (1 bit): A bit that specifies whether the presentation slide can be manually advanced by the user during the slide show.

B - reserved1 (1 bit): MUST be zero and MUST be ignored.

C - fHidden (1 bit): A bit that specifies whether the *corresponding slide* is hidden and is not displayed during the slide show.

D - reserved2 (1 bit): MUST be zero and MUST be ignored.

E - fSound (1 bit): A bit that specifies whether to play the sound specified by **soundIfRef**.

F - reserved3 (1 bit): MUST be zero and MUST be ignored.

G - fLoopSound (1 bit): A bit that specifies whether the sound specified by **soundIdRef** is looped continuously when playing until the next sound plays.

H - reserved4 (1 bit): MUST be zero and MUST be ignored.

I - fStopSound (1 bit): A bit that specifies whether to stop any currently playing sound when the transition starts.

J - reserved5 (1 bit): MUST be zero and MUST be ignored.

K - fAutoAdvance (1 bit): A bit that specifies whether the slide will automatically advance after **slideTime** milliseconds during the slide show.

L - reserved6 (1 bit): MUST be zero and MUST be ignored.

M - fCursorVisible (1 bit): A bit that specifies whether to display the cursor during the slide show.

N - reserved7 (3 bits): MUST be zero and MUST be ignored.

speed (1 byte): A byte value that specifies how long the transition takes to run.

Value	Meaning
0x00	0.75 seconds
0x01	0.5 seconds
0x02	0.25 seconds

unused (3 bytes): Undefined and MUST be ignored.

2.6.7 InteractiveInfoInstance

Referenced by: SlideListWithTextSubContainerOrAtom, TextClientDataSubContainerOrAtom

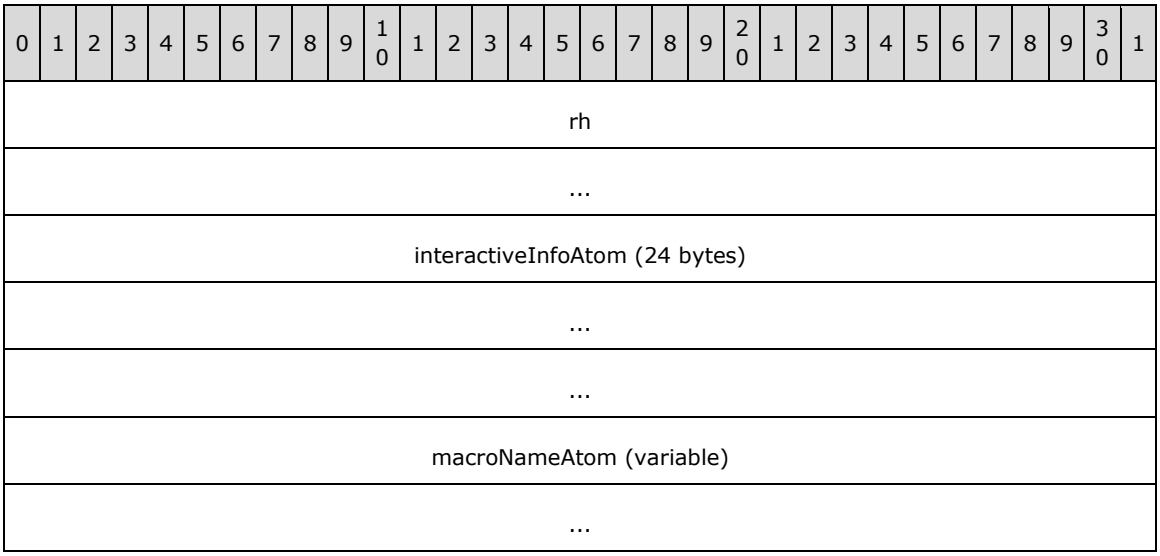
A variable type record whose type and meaning are dictated by the value of **rh.recInstance** as specified in the following table.

Value	Meaning
0x000	A MouseClickInteractiveInfoContainer record that specifies actions to perform when the mouse is clicked on an object.
0x001	A MouseOverInteractiveInfoContainer record that specifies actions to perform when the mouse is moved over an object.

2.6.8 MouseClickInteractiveInfoContainer

Referenced by: InteractiveInfoInstance, OfficeArtClientData

A container record that specifies what actions to perform when interacting with an object by means of a mouse click.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_InteractiveInfo.

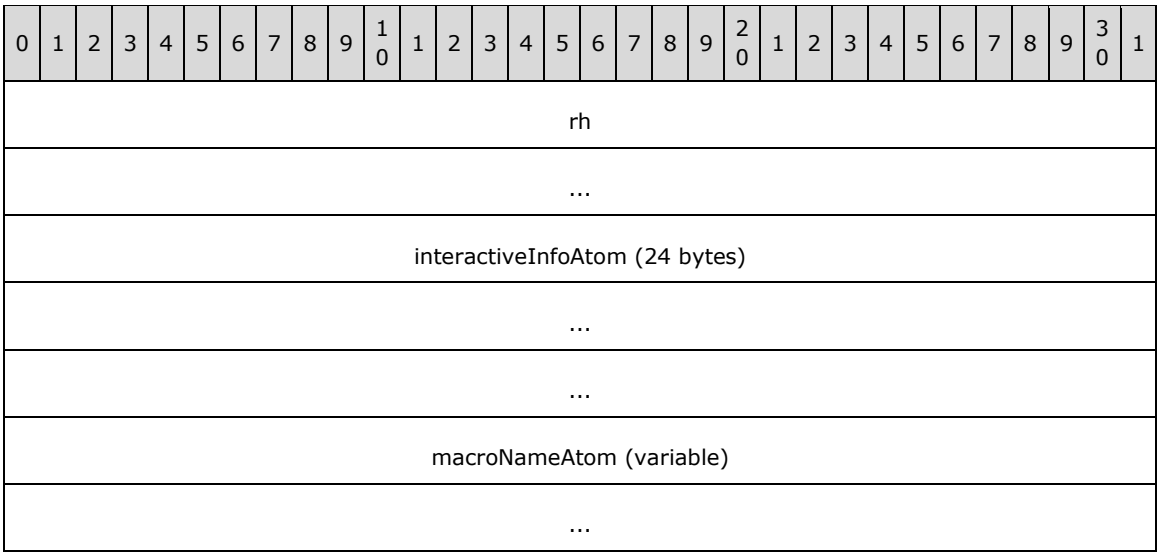
interactiveInfoAtom (24 bytes): An InteractiveInfoAtom record that specifies the type of action to be performed.

macroNameAtom (variable): An optional MacroNameAtom record that specifies the name of a **macro**, a file name, or a named show. It MUST be ignored unless **interactiveInfoAtom.action** is equal to "II_MacroAction", "II_RunProgramAction", or "II_CustomShowAction".

2.6.9 MouseOverInteractiveInfoContainer

Referenced by: InteractiveInfoInstance, OfficeArtClientData

A container record that specifies what actions to perform when interacting with an object by moving the mouse cursor over it.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be RT_InteractiveInfo.

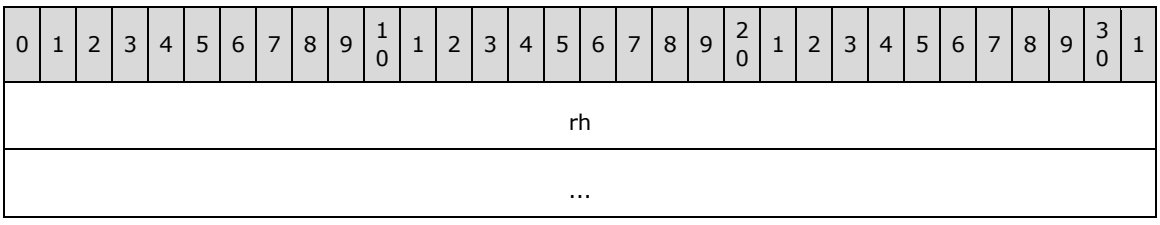
interactiveInfoAtom (24 bytes): An InteractiveInfoAtom record that specifies the type of action to be performed.

macroNameAtom (variable): An optional MacroNameAtom record that specifies the name of a macro, a file name, or a named show. It MUST be ignored unless **interactiveInfoAtom.action** is equal to "II_MacroAction", "II_RunProgramAction", or "II_CustomShowAction".

2.6.10 InteractiveInfoAtom

Referenced by: MouseClickInteractiveInfoContainer, MouseOverInteractiveInfoContainer

An atom record that specifies a type of action to be performed.



soundIdRef							
exHyperlinkIdRef							
action	oleVerb	jump	A	B	C	D	E
hyperlinkType	unused						

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_InteractiveInfoAtom.
rh.recLen	MUST be 0x00000010.

soundIdRef (4 bytes): A SoundIdRef that specifies the sound to play when this action executes.

exHyperlinkIdRef (4 bytes): An ExHyperlinkIdRef that specifies the hyperlink to follow when this action executes. It MUST be ignored unless **action** is equal to "II_JumpAction", "II_HyperlinkAction", or "II_CustomShowAction".

action (1 byte): An **InteractiveInfoActionEnum** enumeration that specifies the action to perform when this action executes.

oleVerb (1 byte): An **OLEVerbEnum** enumeration that specifies the OLE verb to run when this action executes. It MUST be ignored unless **action** is equal to "II_OLEAction".

jump (1 byte): An **InteractiveInfoJumpEnum** enumeration that specifies the slide to jump to. It MUST be ignored unless **action** is equal to "II_JumpAction".

A - fAnimated (1 bit): A bit that specifies whether to animate the object this action applies to when the action is performed.

B - fStopSound (1 bit): A bit that specifies whether to stop currently playing sounds. It MUST be ignored if the **soundIdRef** field specifies a sound to play.

C - fCustomShowReturn (1 bit): A bit that specifies to return to the previous set of displayed slides at the end of the named show. It MUST be ignored unless **action** is equal to II_CustomShowAction.

D - fVisited (1 bit): A bit that specifies whether this action was executed since the file was last loaded.

E - reserved (4 bits): MUST be zero and MUST be ignored.

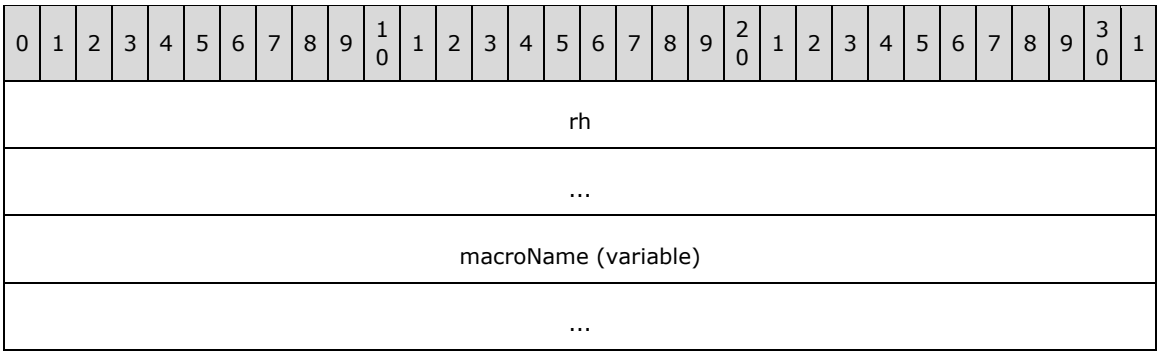
hyperlinkType (1 byte): A **LinkToEnum** enumeration that specifies how to interpret the hyperlink referred to by **exHyperlinkIdRef**.

unused (3 bytes): Undefined and MUST be ignored.

2.6.11 MacroNameAtom

Referenced by: MouseClickInteractiveInfoContainer, MouseOverInteractiveInfoContainer

An atom record that specifies the name of a macro, a file name, or a named show.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

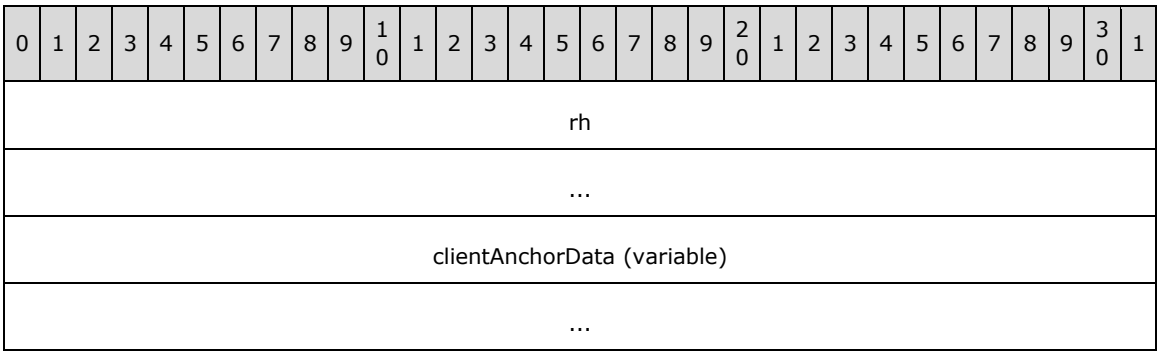
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x002.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number.

macroName (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the name of a macro, a file name, or a named show. If this field specifies a macro and it is not the same as a subroutine present in the VBA project, it MUST be ignored. The length, in bytes, of the field is specified by **rh.recLen**.

2.7 Shape Types

2.7.1 OfficeArtClientAnchor

An atom record that specifies the location of a shape.



rh (8 bytes): An **OfficeArtRecordHeader** ([MS-ODRAW] section 2.2.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be 0xF010.

Field	Meaning
rh.recLen	MUST be 0x00000008 or 0x00000010.

clientAnchorData (variable): An OfficeArtClientAnchorData structure that specifies the location.

2.7.2 OfficeArtClientAnchorData

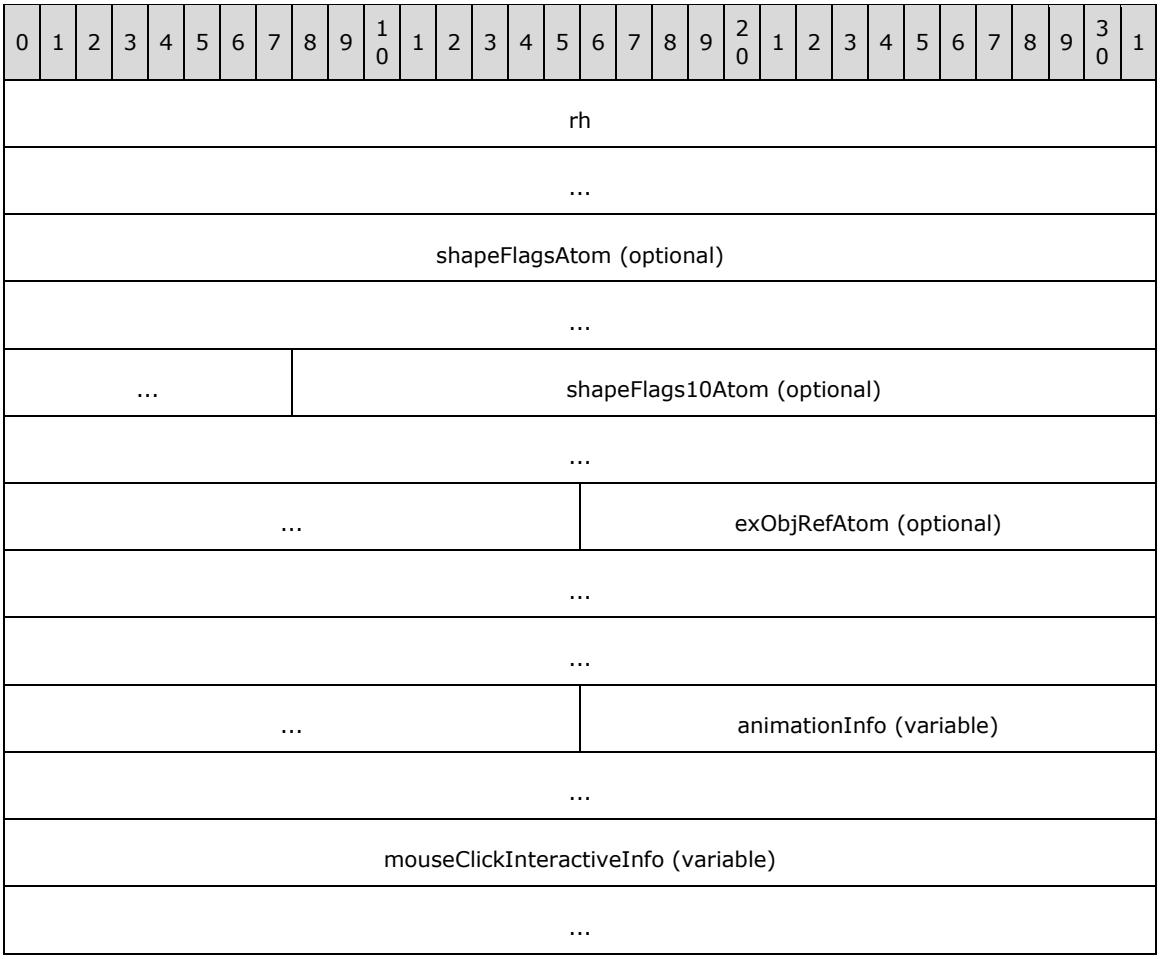
Referenced by: OfficeArtClientAnchor

A variable type structure whose type and meaning are dictated by the value of **rh.recLen** of the OfficeArtClientAnchor record that contains this **OfficeArtClientAnchorData** structure, as specified in the following table.

Value	Meaning
0x00000008	A SmallRectStruct structure that specifies the location of the shape.
0x00000010	A RectStruct structure that specifies the location of the shape.

2.7.3 OfficeArtClientData

A container record that specifies information about a shape.



mouseOverInteractiveInfo (variable)
...
placeholderAtom (16 bytes, optional)
...
...
recolorInfoAtom (variable)
...
rgShapeClientRoundtripData (variable)
...

rh (8 bytes): An **OfficeArtRecordHeader** ([MS-ODRAW] section 2.2.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be 0xF011.

shapeFlagsAtom (9 bytes): An optional ShapeFlagsAtom record that specifies flags for the shape.

shapeFlags10Atom (9 bytes): An optional ShapeFlags2.7.6Atom record that specifies flags for the shape.

exObjRefAtom (12 bytes): An optional ExObjRefAtom record that specifies a reference to an external object.

animationInfo (variable): An optional AnimationInfoContainer record that specifies animation information for the shape.

mouseClickInteractiveInfo (variable): An optional MouseClickInteractiveInfoContainer record that specifies information about interacting with the shape by clicking the mouse on the shape.

mouseOverInteractiveInfo (variable): An optional MouseOverInteractiveInfoContainer record that specifies information about interacting with the shape by moving the mouse over the shape.

placeholderAtom (16 bytes): An optional PlaceholderAtom record that specifies whether the shape is a placeholder shape.

recolorInfoAtom (variable): An optional RecolorInfoAtom record that specifies a collection of re-color mappings for the shape.

rgShapeClientRoundtripData (variable): An array of ShapeClientRoundtripDataSubContainerOrAtom records that specifies additional information about a shape. The array continues while **rh.recType** of the ShapeClientRoundtripDataSubContainerOrAtom item is equal to one of the following record types:

RT_ProgTags, RT_RoundTripNewPlaceholderId12Atom, RT_RoundTripShapeId12Atom, RT_RoundTripHFPlaceholder12Atom, or RT_RoundTripShapeCheckSumForCL12Atom. Each record type MUST NOT appear more than once.

2.7.4 ShapeClientRoundtripDataSubContainerOrAtom

Referenced by: OfficeArtClientData

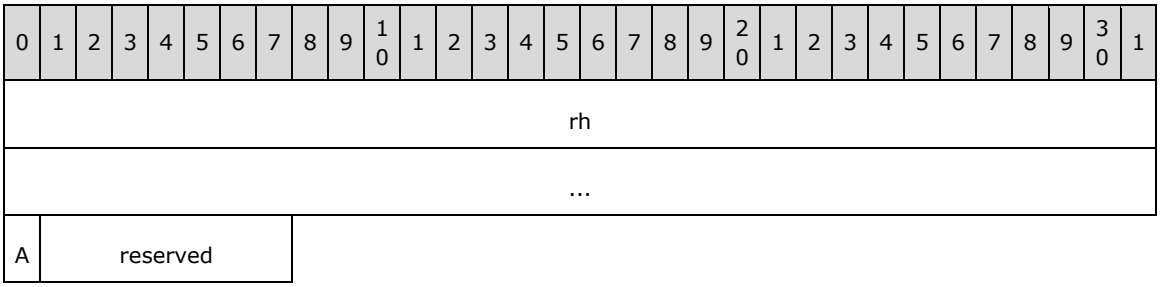
A variable type record whose type and meaning are dictated by the value of **rh.recType**, as specified by the following table.

Value	Meaning
RT_ProgTags (section 2.13.24)	A ShapeProgTagsContainer record that specifies programmable tags for the shape.
RT_RoundTripNewPlaceholderId12Atom	A RoundTripNewPlaceholderId2.11.17Atom record that specifies a placeholder shape identifier. It SHOULD<88> be ignored and SHOULD<89> be preserved.
RT_RoundTripShapeId12Atom	A RoundTripShapeId2.11.22Atom record that specifies a shape identifier. It SHOULD<90> be ignored and SHOULD<91> be preserved.
RT_RoundTripHFPlaceholder12Atom	A RoundTripHFPlaceholder2.11.16Atom record that specifies whether a shape is a header or footer placeholder shape. It SHOULD<92> be ignored and SHOULD<93> be preserved.
RT_RoundTripShapeCheckSumForCL12Atom	A RoundTripShapeCheckSumForCustomLayouts2.11.21Atom record that specifies checksum values for a shape**.** It SHOULD<94> be ignored and SHOULD<95> be preserved.

2.7.5 ShapeFlagsAtom

Referenced by: OfficeArtClientData

An atom record that specifies shape-level **Boolean** flags. More flags are specified in the ShapeFlags2.7.6Atom record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_ShapeAtom.
rh.recLen	MUST be 0x00000001.

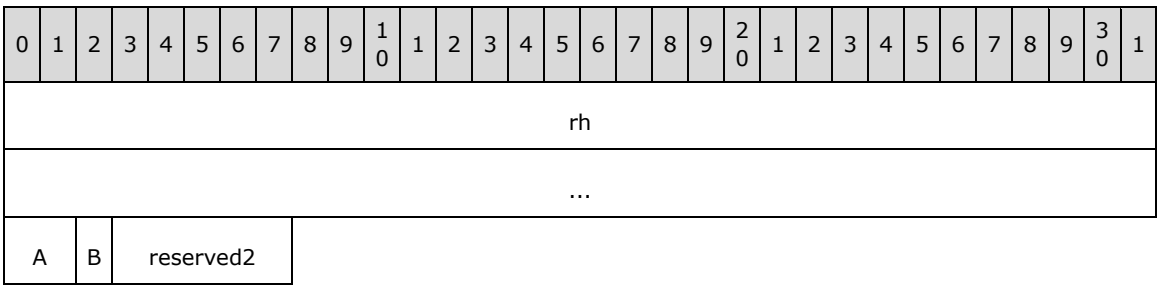
A - fAlwaysOnTop (1 bit): A bit that specifies whether a shape is rendered on top of other shapes.

reserved (7 bits): MUST be zero and MUST be ignored.

2.7.6 ShapeFlags10Atom

Referenced by: OfficeArtClientData

An atom record that specifies shape-level **Boolean** flags. More flags are specified in the ShapeFlagsAtom record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_ShapeFlags10Atom.
rh.recLen	MUST be 0x00000001.

A - reserved1 (2 bits): MUST be zero and MUST be ignored.

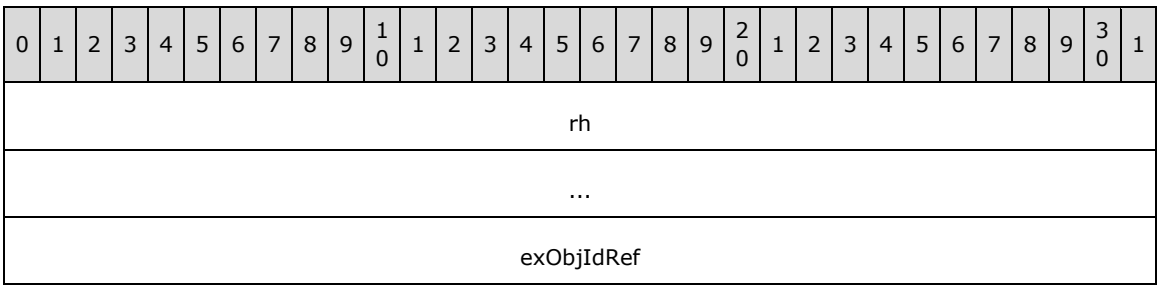
B - fIsPhotoAlbumPicture (1 bit): A bit that specifies whether a shape is a picture in a photo album specified by the PhotoAlbumInfo2.4.9Atom record. It MAY<96> be ignored.

reserved2 (5 bits): MUST be zero and MUST be ignored.

2.7.7 ExObjRefAtom

Referenced by: OfficeArtClientData

An atom record that specifies a reference to an external object.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_ExternalObjectRefAtom.
rh.recLen	MUST be 0x00000004.

exObjIdRef (4 bytes): An ExObjIdRef that specifies an identifier that references an external object.

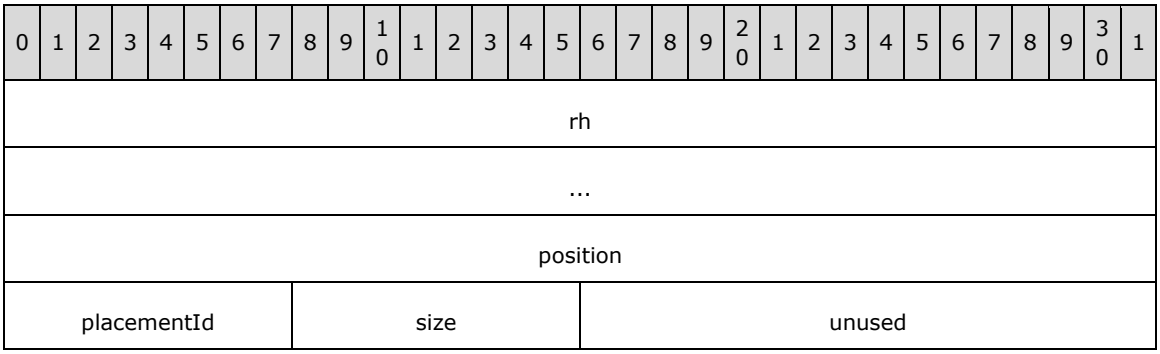
2.7.8 PlaceholderAtom

Referenced by: OfficeArtClientData

An atom record that specifies whether a shape is a placeholder shape. The number, position, and type of placeholder shapes are determined by the slide layout as specified in the SlideAtom record.

Let the *corresponding shape* be specified by the **OfficeArtSpContainer** record ([MS-ODRAW] section 2.2.14) that contains this **PlaceholderAtom** record.

Let the *corresponding slide* be specified by the **MainMasterContainer** record (section 2.5.3), **HandoutContainer** record (section 2.5.8), **SlideContainer** record (section 2.5.1), or **NotesContainer** record (section 2.5.6) that contains this **PlaceholderAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_PlaceholderAtom.
rh.recLen	MUST be 0x00000008.

position (4 bytes): A **signed integer** that specifies an identifier for the placeholder shape. It SHOULD be unique among all **PlaceholderAtom** records contained in the *corresponding slide*. The value 0xFFFFFFFF specifies that the *corresponding shape* is not a placeholder shape.

placementId (1 byte): A **PlaceholderEnum** enumeration that specifies the type of the placeholder shape. The value MUST conform to the constraints as specified in the following table.

Value	Meaning
PT_None	MUST NOT be used for this field.
PT_MasterTitle	The *corresponding shape* contains the master title text. The *corresponding slide* MUST be a main master slide.
PT_MasterBody	The *corresponding shape* contains the master body text. The *corresponding slide* MUST be a main master slide.
PT_MasterCenterTitle	The *corresponding shape* contains the master center title text. The *corresponding slide* MUST be a title master slide.
PT_MasterSubTitle	The *corresponding shape* contains the master sub-title text. The *corresponding slide* MUST be a title master slide.
PT_MasterNotesSlideImage	The *corresponding shape* contains the shared properties for slide image shapes. The *corresponding slide* MUST be a notes master slide.
PT_MasterNotesBody	The *corresponding shape* contains the master body text. The *corresponding slide* MUST be a notes master slide.
PT_MasterDate	The *corresponding shape* contains the date text field. The *corresponding slide* MUST be a main master slide, title master slide, notes master slide, or handout master slide.
PT_MasterSlideNumber	The *corresponding shape* contains a slide number text field. The *corresponding slide* MUST be a main master slide, title master slide, notes master slide, or handout master slide.
PT_MasterFooter	The *corresponding shape* contains a footer text field. The *corresponding slide* MUST be a main master slide, title master slide, notes master slide, or handout master slide.
PT_MasterHeader	The *corresponding shape* contains a header text field. The *corresponding slide* must be a notes master slide or handout master slide.
PT_NotesSlideImage	The *corresponding shape* contains a presentation slide image. The *corresponding slide* MUST be a notes slide.
PT_NotesBody	The *corresponding shape* contains the notes text. The *corresponding slide* MUST be a notes slide.
PT_Title	The *corresponding shape* contains the title text. The *corresponding slide* MUST be a presentation slide.
PT_Body	The *corresponding shape* contains the body text. The *corresponding slide* MUST be a presentation slide.
PT_CenterTitle	The *corresponding shape* contains the center title text. The *corresponding slide* MUST be a presentation slide.
PT_SubTitle	The *corresponding shape* contains the sub-title text. The *corresponding slide* MUST be a presentation slide.
PT_VerticalTitle	The *corresponding shape* contains the title text with vertical text flow. The corresponding slide MUST be a presentation slide.
PT_VerticalBody	The *corresponding shape* contains the body text with vertical text flow. The corresponding slide MUST be a presentation slide.

Value	Meaning
PT_Object	The *corresponding shape* contains a generic object. The *corresponding slide* MUST be a presentation slide.
PT_Graph	The *corresponding shape* contains a chart object. The *corresponding slide* MUST be a presentation slide.
PT_Table	The *corresponding shape* contains a table object. The *corresponding slide* MUST be a presentation slide.
PT_ClipArt	The *corresponding shape* contains a clipart object. The *corresponding slide* MUST be a presentation slide.
PT_OrgChart	The *corresponding shape* contains an organization chart object. The corresponding slide MUST be a presentation slide.
PT_Media	The *corresponding shape* contains a media object. The *corresponding slide* MUST be a presentation slide.
PT_VerticalObject	The *corresponding shape* contains a generic object with vertical text flow. The *corresponding slide* MUST be a presentation slide.
PT_Picture	The *corresponding shape* contains a picture object. The *corresponding slide* MUST be a presentation slide.

size (1 byte): A **PlaceholderSize** enumeration that specifies the preferred size of the placeholder shape.

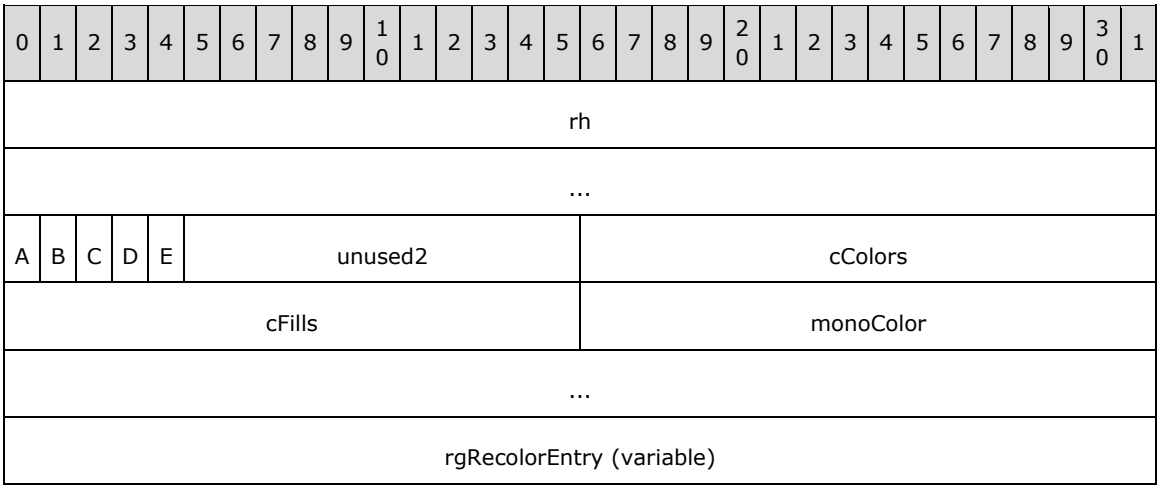
unused (2 bytes): Undefined and MUST be ignored.

2.7.9 RecolorInfoAtom

Referenced by: OfficeArtClientData

An atom record that specifies a collection of re-color mappings for a **metafile** ([MS-WMF]).

The *corresponding metafile* is specified by the **Blip** properties ([MS-ODRAW] section 2.3.23) of the **OfficeArtSpContainer** ([MS-ODRAW] section 2.2.14) that contains this **RecolorInfoAtom** record.



...
unused3 (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_RecolorInfoAtom.

A - fShouldRecolor (1 bit): A bit that specifies whether the re-color mappings are applied.

B - fMissingColors (1 bit): A bit that specifies whether **rgRecolorEntry** has more than 64 items with a RecolorEntryColor variant. It SHOULD<97> be ignored.

C - fMissingFills (1 bit): A **bit** that specifies whether **rgRecolorEntry** has more than 64 items with a RecolorEntryBrush variant. It SHOULD<98> be ignored.

D - unused1 (1 bit): Undefined and MUST be ignored.

E - fMonoRecolor (1 bit): A bit that specifies whether **monoColor** is used as the destination color instead of the destination colors specified by the RecolorEntry structures in the **rgRecolorEntry** array.

unused2 (11 bits): Undefined and MUST be ignored.

cColors (2 bytes): An unsigned integer that specifies the count of items in the **rgRecolorEntry** array with a RecolorEntryColor variant.

cFills (2 bytes): An unsigned integer that specifies the count of items in the **rgRecolorEntry** array with a RecolorEntryBrush variant.

monoColor (6 bytes): A WideColorStruct structure that specifies the destination color if the **fMonoRecolor** bit is set.

rgRecolorEntry (variable): An array of RecolorEntry structures that specifies color mappings. The count of items in the array is specified by **cColors** + **cFills**. It MUST contain **cColor** items with a RecolorEntryColor variant. It MUST contain **cFills** items with a RecolorEntryBrush variant.

unused3 (variable): Undefined and MUST be ignored. The size, in bytes, is specified by the following formula:

```
rh.recLen - ( 12 + 44 * ( cColors + cFills ) )
```

2.7.10 RecolorEntry

Referenced by: RecolorInfoAtom

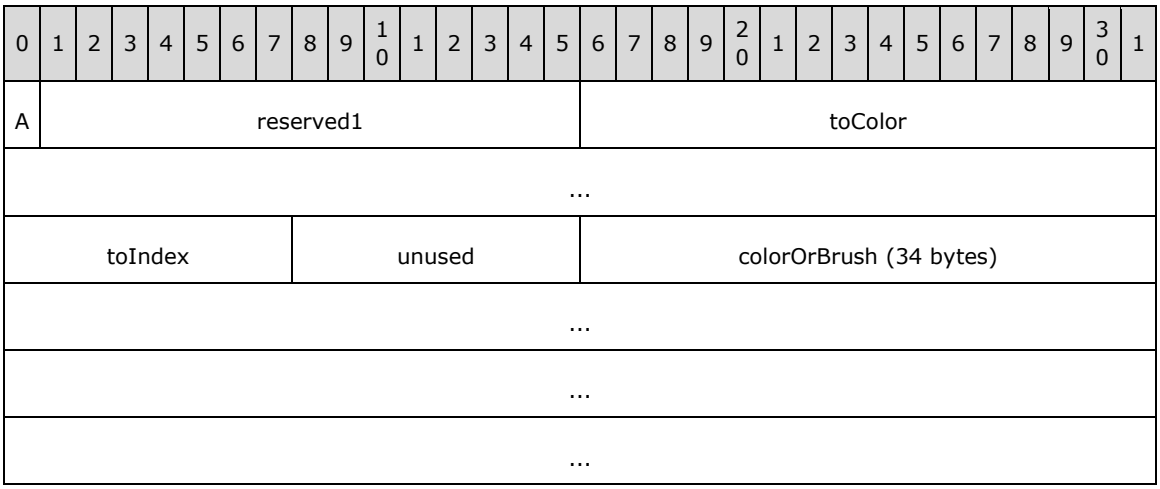
A structure that specifies a color mapping for metafile records. A color mapping has a source color description and a destination color description. Applying a color mapping to a metafile means to be the

same as the source color description with the corresponding fields of the listed metafile records that follow and replace them with metafile records which represent the mapped destination color.

The following metafile ([MS-WMF]) records are the targets for possible color replacement.

Metafile record	Specified in
META_SETTEXTCOLOR	[MS-WMF] section 2.3.5.26
META_SETBKCOLOR	[MS-WMF] section 2.3.5.14
META_CREATEPENINDIRECT	[MS-WMF] section 2.3.4.5
META_CREATEBRUSHINDIRECT	[MS-WMF] section 2.3.4.1
META_CREATEPATTERNBRUSH	[MS-WMF] section 2.3.4.4
META_DIBCREATEPATTERNBRUSH	[MS-WMF] section 2.3.4.8
META_DIBSTRETCHBLT	[MS-WMF] section 2.3.1.3
META_STRETCHDIB	[MS-WMF] section 2.3.1.6

Let the *corresponding color scheme* be as specified by the **MainMasterContainer** record (section 2.5.3), **HandoutContainer** record (section 2.5.8), **SlideContainer** record (section 2.5.1), or **NotesContainer** record (section 2.5.6) that contains this **RecolorEntry** structure.



A - fDoRecolor (1 bit): A bit that specifies whether the color mapping is performed.

reserved1 (15 bits): MUST be zero and MUST be ignored.

toColor (6 bytes): A WideColorStruct structure that specifies the destination color of the mapping when **toIndex** is greater than or equal to 8. It MUST be ignored if **toIndex** is less than 8.

toIndex (1 byte): An unsigned integer that specifies the destination color of the mapping. If the value is less than 8, it is a 0-based index into the *corresponding color scheme*. If the value is greater than or equal to 8, **toColor** is used for the destination color.

unused (1 byte): Undefined and MUST be ignored.

colorOrBrush (34 bytes): A RecolorEntryVariant structure that specifies the source color of the color mapping.

2.7.11 RecolorEntryVariant

Referenced by: RecolorEntry

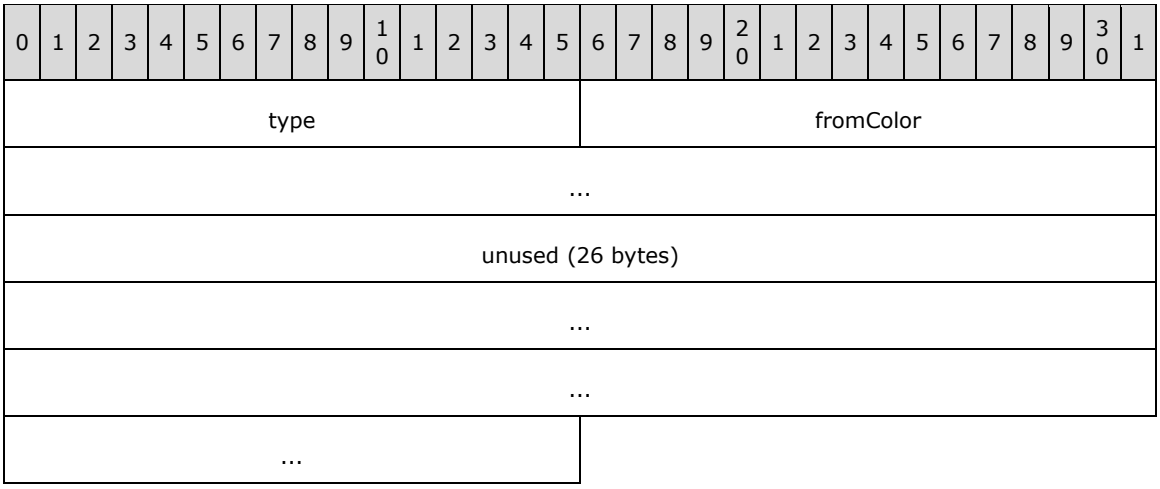
A variant structure whose type and meaning are dictated by the value of the **type** field of either of these two structures, as specified in the following table.

Type	Meaning
0x0000	A RecolorEntryColor structure that specifies a source color.
0x0001	A RecolorEntryBrush structure that specifies a source brush.

2.7.12 RecolorEntryColor

Referenced by: RecolorEntryVariant

A structure that specifies a source color for a RecolorEntry structure.



type (2 bytes): An **unsigned integer** that specifies the variant of the containing RecolorEntryVariant structure. It MUST be 0x0000.

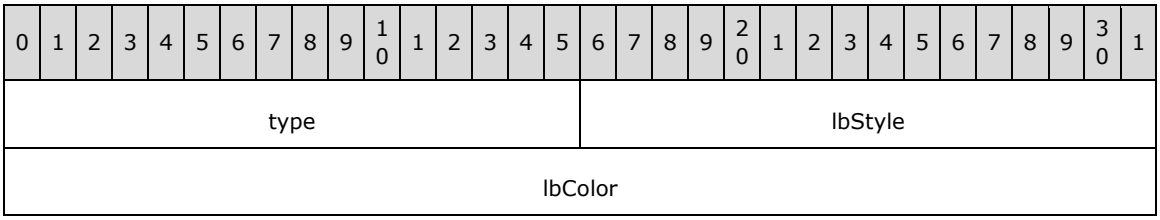
fromColor (6 bytes): A WideColorStruct structure that specifies a source color.

unused (26 bytes): Undefined and MUST be ignored.

2.7.13 RecolorEntryBrush

Referenced by: RecolorEntryVariant

A structure that specifies a source color for a RecolorEntry that corresponds to a **LogBrush** Object as specified in [MS-WMF] section 2.2.2.10. The meaning of the fields corresponds to the meaning of the fields of a **LogBrush** Object.



...	lbHatch
fgColor	
...	bgColor
...	
bitmapType	pattern
...	
...	

type (2 bytes): An unsigned integer that specifies the variant of the containing RecolorEntryVariant structure. It MUST be 0x0001.

lbStyle (2 bytes): An unsigned integer that specifies a brush type. It MUST be a **BrushStyle Enumeration** as specified in [MS-WMF] section 2.1.1.4.

lbColor (6 bytes): A WideColorStruct structure that specifies the color of the **LogBrush** Object. Its interpretation depends on the value of **lbStyle** and is specified in [MS-WMF] section 2.2.2.10.

lbHatch (2 bytes): An unsigned integer that specifies a brush hatch type. Its interpretation depends on the value of **lbStyle** and is specified in [MS-WMF] section 2.1.1.12.

fgColor (6 bytes): A WideColorStruct structure that specifies a foreground color. Only used for metafile records of type META_DIBCREATEPATTERNBRUSH, specified in [MS-WMF] section 2.3.4.8. This field represents the color of the first entry of the **DIB** color table, specified in [MS-WMF] section 2.2.2.3. Undefined and MUST be ignored if **lbStyle** is not equal to 0x0003.

bgColor (6 bytes): A WideColorStruct structure that specifies a background color. Only used for metafile records of type META_DIBCREATEPATTERNBRUSH, as specified in [MS-WMF] section 2.3.4.8. This field represents the color of the second entry of the **DIB** color table, as specified in [MS-WMF] section 2.2.2.3. Undefined and MUST be ignored if **lbStyle** is not equal to 0x0003.

bitmapType (2 bytes): An unsigned integer that specifies the type of the bitmap if **lbStyle** is equal to 0x0003. It MUST also be a value from the following table.

Value	Meaning
0x0000	Color mapping is used for META_CREATEPATTERNBRUSH records with a monochrome pattern bitmap.
0x0001	Color mapping is used for META_DIBCREATEPATTERNBRUSH records.
0x0003	Color mapping is used for META_CREATEBRUSHINDIRECT records and META_CREATEPATTERNBRUSH records with a non-monochrome pattern bitmap.

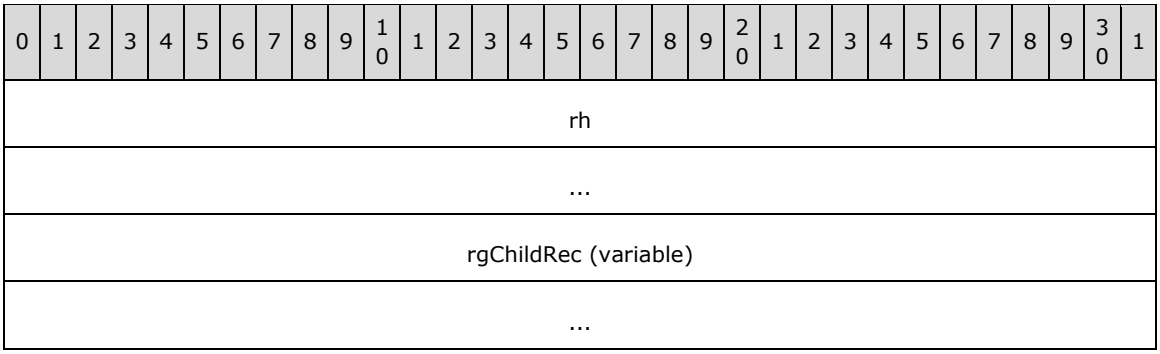
MUST be ignored if **lbStyle** is not equal to 0x0003.

pattern (8 bytes): An array of bytes that specifies the bit pattern for a monochrome 8x8 pixel brush. Undefined and MUST be ignored if **lbStyle** is not equal to 0x0003 or if **bitmapType** is equal to 0x0003. If **bitmapType** is equal to 0x0000 it specifies the **Bits** field for a META_CREATEPATTERNBRUSH record as specified in [MS-WMF] section 2.3.4.4. If **bitmapType** is equal to 0x0001 it specifies the **Bits** field of a DeviceIndependentBitmap (DIB) Object ([MS-WMF] section 2.2.2.3) for a META_DIBCREATEPATTERNBRUSH record as specified in [MS-WMF] sections 2.3.4.8.

2.7.14 ShapeProgTagsContainer

Referenced by: ShapeClientRoundtripDataSubContainerOrAtom

A container record that specifies programmable tags with additional shape data.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	SHOULD<99> be 0x000.
rh.recType	MUST be **RT_ProgTags** (section 2.13.24).

rgChildRec (variable): An array of ShapeProgTagsSubContainerOrAtom records that specifies the programmable tags. The size, in bytes, of the array is specified by **rh.recLen**. The array MUST NOT contain more than one of each of the following records: PP2.7.18ShapeBinaryTagExtension, PP2.7.19ShapeBinaryTagExtension, or PP2.7.20ShapeBinaryTagExtension.

2.7.15 ShapeProgTagsSubContainerOrAtom

Referenced by: ShapeProgTagsContainer

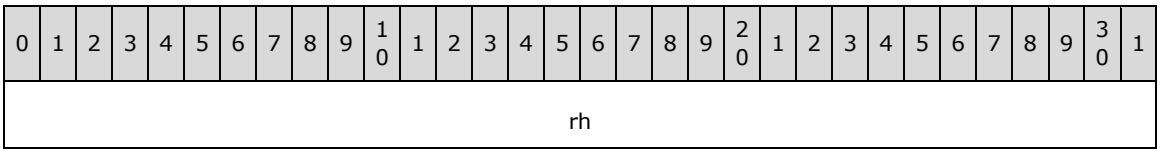
A variable type record whose type and meaning are dictated by the value of **rh.recType**, as specified in the following table.

Value	Meaning
RT_ProgStringTag	A ProgStringTagContainer record that specifies additional shape data.
RT_ProgBinaryTag	A ShapeProgBinaryTagContainer record that specifies additional shape data.

2.7.16 ShapeProgBinaryTagContainer

Referenced by: ShapeProgTagsSubContainerOrAtom

A container record that specifies programmable tags with additional binary shape data.



...
rec (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_ProgBinaryTag.

rec (variable): A ShapeProgBinaryTagSubContainerOrAtom record that specifies additional shape data.

2.7.17 ShapeProgBinaryTagSubContainerOrAtom

Referenced by: ShapeProgBinaryTagContainer

A variable type record whose type and meaning are dictated by the value of **tagNameAtom.tagName** for UnknownBinaryTag or by the value of **tagName** for PP2.7.18ShapeBinaryTagExtension, PP2.7.19ShapeBinaryTagExtension, PP2.7.20ShapeBinaryTagExtension, as specified in the following table.

Value	Meaning
"___PPT9"	A PP2.7.18ShapeBinaryTagExtension record pair that specifies additional shape data. It MAY<100> be ignored and MUST be preserved.
"___PPT10"	A PP2.7.19ShapeBinaryTagExtension record pair that specifies additional shape data. It MAY<101> be ignored and MUST be preserved.
"___PPT11"	A PP2.7.20ShapeBinaryTagExtension record pair that specifies additional shape data. It MAY<102> be ignored and MUST be preserved.
Any other value	An UnknownBinaryTag record pair that specifies additional shape data. It MUST be ignored and MUST be preserved.

2.7.18 PP9ShapeBinaryTagExtension

Referenced by: ShapeProgBinaryTagSubContainerOrAtom

A pair of atom records that specifies a programmable tag with additional shape data.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															

tagName (14 bytes)	
...	
...	
...	rhData
...	
...	styleTextPropAtom (variable)
...	

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the first record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be 0x0000000E.

tagName (14 bytes): A **PrintableUnicodeString** (section 2.2.23) that specifies the programmable tag name. It MUST be "___PPT9".

rhData (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the second record. Sub-fields are further specified in the following table:

Field	Meaning
rhData.recVer	MUST be 0x0.
rhData.recInstance	MUST be 0x000.
rhData.recType	MUST be RT_BinaryTagDataBlob.

styleTextPropAtom (variable): A StyleTextProp2.9.67Atom record that specifies additional text style properties.

2.7.19 PP10ShapeBinaryTagExtension

Referenced by: ShapeProgBinaryTagSubContainerOrAtom

A pair of atom records that specifies a programmable tag with additional shape data.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															

tagName (16 bytes)
...
...
rhData
...
styleTextPropAtom (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the first record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be 0x00000010.

tagName (16 bytes): A **PrintableUnicodeString** (section 2.2.23) that specifies the programmable tag name. It MUST be "___PPT10".

rhData (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the second record. Sub-fields are further specified in the following table.

Field	Meaning
rhData.recVer	MUST be 0x0.
rhData.recInstance	MUST be 0x000.
rhData.recType	MUST be RT_BinaryTagDataBlob.

styleTextPropAtom (variable): A StyleTextProp2.9.69Atom record that specifies additional text style properties.

2.7.20 PP11ShapeBinaryTagExtension

Referenced by: ShapeProgBinaryTagSubContainerOrAtom

A pair of atom records that specifies a programmable tag with additional shape data.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															

tagName (16 bytes)
...
...
rhData
...
styleTextPropAtom (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be **RT_CString** (section 2.13.24).
rh.recLen	MUST be 0x00000010.

tagName (16 bytes): A **PrintableUnicodeString** (section 2.2.23) that specifies the programmable tag name. It MUST be "___PPT11".

rhData (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for the second record. Sub-fields are further specified in the following table.

Field	Meaning
rhData.recVer	MUST be 0x0.
rhData.recInstance	MUST be 0x000.
rhData.recType	MUST be RT_BinaryTagDataBlob.

styleTextPropAtom (variable): A StyleTextProp2.9.70Atom record that specifies additional text style properties.

2.8 Animation Types

2.8.1 AnimationInfoContainer

Referenced by: OfficeArtClientData

A container record that specifies the animation and sound information for a shape.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															

...
animationAtom (36 bytes)
...
...
animationSound (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_AnimationInfo.

animationAtom (36 bytes): An AnimationInfoAtom record that specifies the animation effect information for the shape. It SHOULD<103> be ignored.

animationSound (variable): An optional **SoundContainer** record (section 2.4.16.3) that specifies the sound for the animation specified by the **animationAtom**. If this field exists, it overrides the **animationAtom.soundIdRef**.

2.8.2 AnimationInfoAtom

Referenced by: AnimationInfoContainer

An atom record that specifies the animation information for a shape or text.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
dimColor																															
A		B		C		D		E		F		G		H		reserved															
soundIdRef																															
delayTime																															
orderID																															
slideCount																															

animBuildType	animEffect	animEffectDirection	animAfterEffect
textBuildSubEffect	oleVerb	unused	

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x1.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_AnimationInfoAtom.
rh.recLen	MUST be 0x0000001C.

dimColor (4 bytes): A ColorIndexStruct structure that specifies a color for the dim effect after the animation is complete.

A - fReverse (2 bits): An **unsigned integer** that specifies whether the animation plays in the reverse direction. It MUST be a value from the following table.

Value	Meaning
0x0	Do not play in the reverse direction.
0x1	Play in the reverse direction.

B - fAutomatic (2 bits): An **unsigned integer** that specifies whether the animation starts automatically. It MUST be a value from the following table.

Value	Meaning
0x0	Start manually by click.
0x1	Start automatically.

C - fSound (2 bits): An **unsigned integer** that specifies whether the animation has an associated sound. It MUST be a value from the following table.

Value	Meaning
0x0	Has no associated sound.
0x1	Has associated sound.

D - fStopSound (2 bits): An **unsigned integer** that specifies whether all playing sounds are stopped when this animation begins. It MUST be a value from the following table.

Value	Meaning
0x0	All playing sounds are not stopped.
0x1	All playing sounds are stopped.

E - fPlay (2 bits): An unsigned integer that specifies whether an associated sound, media or action verb is activated when the shape is clicked. It MUST be a value from the following table.

Value	Meaning
0x0	No behavior happens when the shape is clicked.
0x1	The associated sound, media or action verb plays when the shape is clicked.

F - fSynchronous (2 bits): An unsigned integer that specifies that the animation, while playing, stops other slide show actions. If the shape is a media or OLE object, this field is valid; otherwise, it MUST be ignored. It MUST be a value from the following table.

Value	Meaning
0x0	Do not stop other slide show actions.
0x1	Stop other slide show actions.

G - fHide (2 bits): An unsigned integer that specifies whether the shape is hidden while the animation is not playing. If the shape is a media or OLE object, this field is valid; otherwise, it MUST be ignored. It MUST be a value from the following table.

Value	Meaning
0x0	Do not hide the shape while the animation is not playing.
0x1	Hide the shape while the animation is not playing.

H - fAnimateBg (2 bits): An unsigned integer that specifies whether the background of the shape is animated. It MUST be a value from the following table.

Value	Meaning
0x0	Do not animate the background.
0x1	Animate the background.

reserved (16 bits): MUST be zero, and MUST be ignored.

soundIdRef (4 bytes): A SoundIdRef that specifies the value to refer to in the **SoundCollectionContainer** record (section 2.4.16.1) to locate the embedded audio.

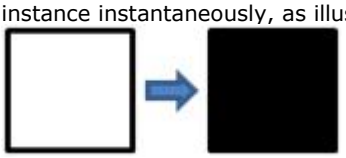
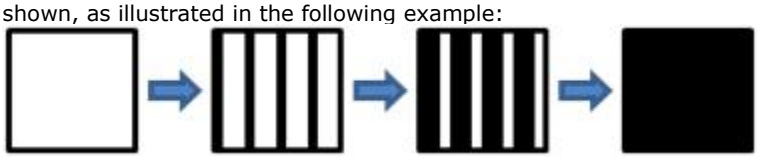
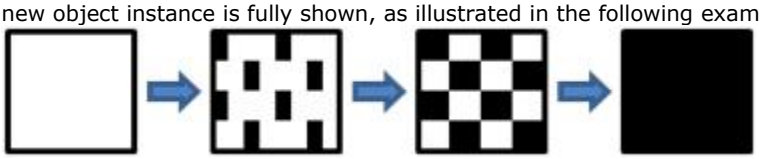
delayTime (4 bytes): A **signed integer** that specifies the delay time, in milliseconds, before the animation starts to play. If **fAutomatic** is 0x1, this value MUST be greater than or equal to 0; otherwise, this field MUST be ignored.

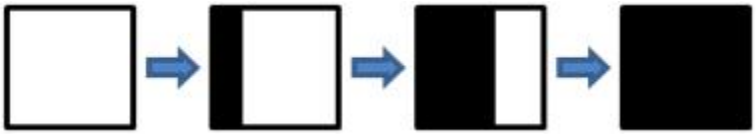
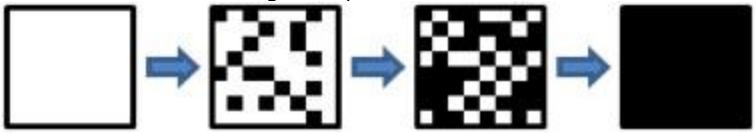
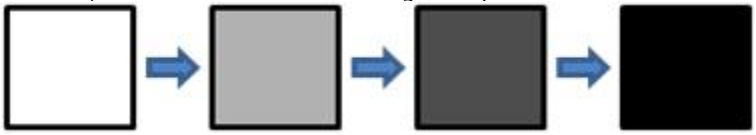
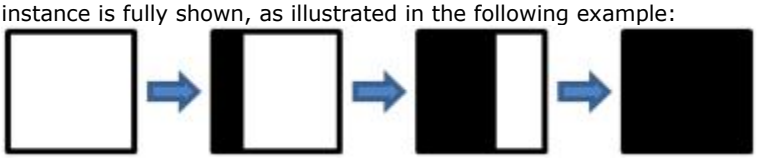
orderID (2 bytes): A signed integer that specifies the order of the animation in the slide. It MUST be greater than or equal to -2. The value -2 specifies that this animation follows the order of the corresponding placeholder shape on the main master slide or title master slide. The value -1 SHOULD NOT<104> be used.

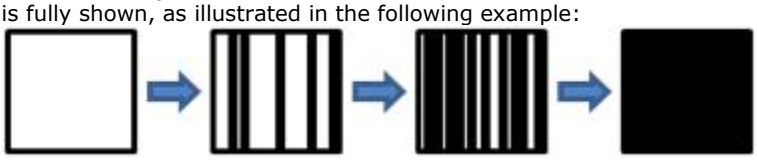
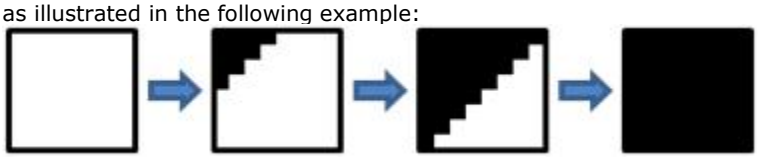
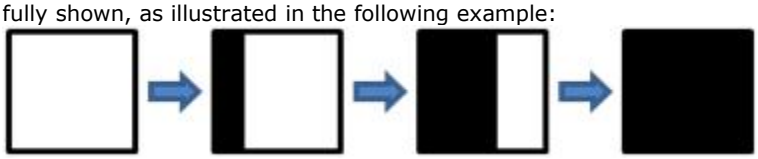
slideCount (2 bytes): An unsigned integer that specifies the number of slides that this animation continues playing. This field is utilized only in conjunction with media. The value 0xFFFFFFFF specifies that the animation plays for one slide.

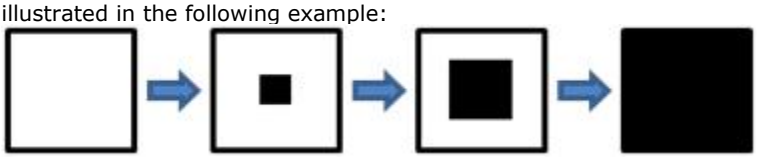
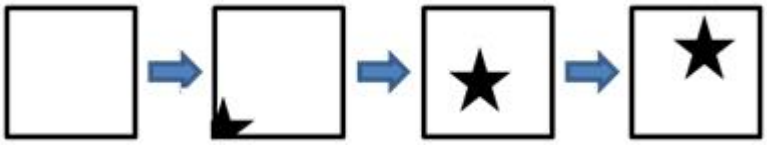
animBuildType (1 byte): An **AnimBuildTypeEnum** enumeration that specifies the animation build type for the animation effect.

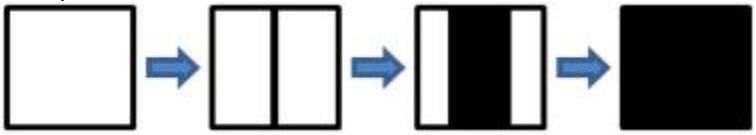
animEffect (1 byte): An unsigned integer that specifies the animation effect type for the shape. The following diagrams are for example purposes only. Exact rendering of any animation effect is determined by the rendering application. As such, the same animation effect can have many variations depending on the implementation. It MUST be a value from the following table:

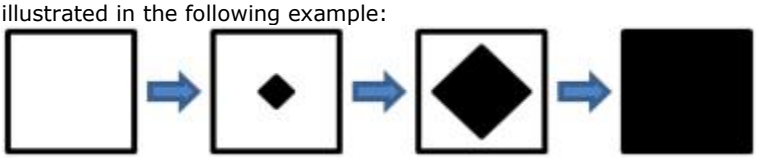
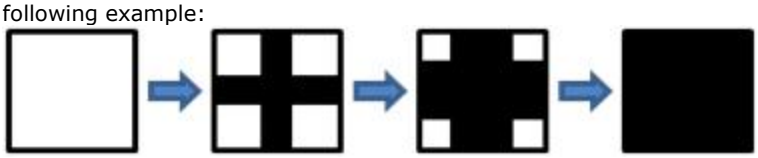
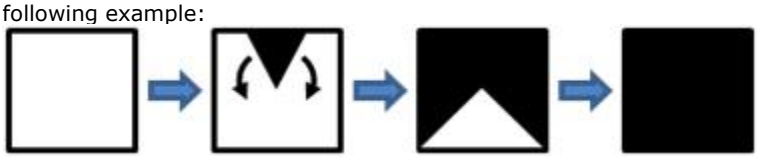
Value	Description
0x00	Cut animation effect that replaces the previous object instance with the new object instance instantaneously, as illustrated in the following example:  The **animEffectDirection** value must be one of the following: ▪ 0x00: Not through black ▪ 0x01: Through black ▪ 0x02: The same as 0x00
0x01	Random animation effect that chooses a random effect with a random applicable direction from the set available. This effect can be different each time it is used. **animEffectDirection** MUST be ignored.
0x02	Blinds animation effect that uses a set of horizontal or vertical bars and wipes them either left-to-right or top-to-bottom, respectively, until the new object instance is fully shown, as illustrated in the following example:  The **animEffectDirection** value must be one of the following: ▪ 0x00: Vertical direction ▪ 0x01: Horizontal direction
0x03	Checker animation effect that uses a set of horizontal or vertical checkerboard squares and wipes them either left-to-right or top-to-bottom, respectively, until the new object instance is fully shown, as illustrated in the following example:  The **animEffectDirection** value must be one of the following: ▪ 0x00: Horizontal direction ▪ 0x01: Vertical direction
0x04	Cover animation effect that moves the new object instance in from the specified direction, continually covering more of the previous object instance until the new object instance is fully shown, as illustrated in the following example:

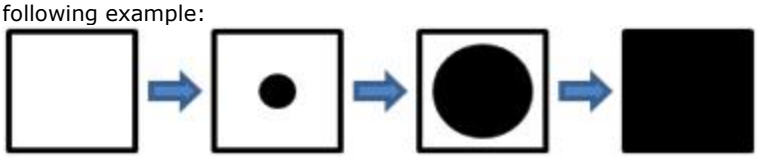
Value	Description
	 The **animEffectDirection** value must be one of the following: ▪ 0x00: Cover from the right to the left side of the object. ▪ 0x01: Cover from the bottom to the top side of the object. ▪ 0x02: Cover from the left to the right side of the object. ▪ 0x03: Cover from the top to the bottom side of the object. ▪ 0x04: Cover from the bottom-right to the top-left corner of the object. ▪ 0x05: Cover from the bottom-left to the top-right corner of the object. ▪ 0x06: Cover from the top-right to the bottom-left corner of the object. ▪ 0x07: Cover from the top-left to the bottom-right corner of the object.
0x05	Dissolve animation effect that uses a set of randomly placed squares on the object instance that continue to be added to until the new object instance is fully shown, as illustrated in the following example:  **animEffectDirection** MUST be 0x00.
0x06	Fade animation effect that smoothly fades the previous object instance either directly to the new object instance or first to a black screen and then to the new object instance, as illustrated in the following example:  **animEffectDirection** MUST be 0x00.
0x07	Pull animation effect that moves the previous object instance out from the specified direction, continually revealing more of the new object instance until the new object instance is fully shown, as illustrated in the following example:  The **animEffectDirection** value must be one of the following: ▪ 0x00: Reveal from the right to the left side of the object. ▪ 0x01: Reveal from the bottom to the top side of the object. ▪ 0x02: Reveal from the left to the right side of the object. ▪ 0x03: Reveal from the top to the bottom side of the object.

Value	Description
	▪ 0x04: Reveal from the bottom-right to the top-left corner of the object. ▪ 0x05: Reveal from the bottom-left to the top-right corner of the object. ▪ 0x06: Reveal from the top-right to the bottom-left corner of the object. ▪ 0x07: Reveal from the top-left to the bottom-right corner of the object.
0x08	Random bar animation effect that uses a set of randomly placed horizontal or vertical bars on the object instance that continue to be added to until the new object instance is fully shown, as illustrated in the following example:  The **animEffectDirection** value must be one of the following: ▪ 0x00: Horizontal direction ▪ 0x01: Vertical direction
0x09	Strips animation effect that uses a set of bars that are arranged in a staggered fashion and wipes them across the screen until the new object instance is fully shown, as illustrated in the following example:  The **animEffectDirection** value must be one of the following: ▪ 0x04: Strips move from the bottom-right to the top-left corner of the object. ▪ 0x05: Strips move from the bottom-left to the top-right corner of the object. ▪ 0x06: Strips move from the top-right to the bottom-left corner of the object. ▪ 0x07: Strips move from the top-left to the bottom-right corner of the object.
0x0A	Wipe animation effect that wipes the new object instance over the previous object instance from one edge of the screen to the opposite until the new object instance is fully shown, as illustrated in the following example:  The **animEffectDirection** value must be one of the following: ▪ 0x00: Wipe effect is from the right to the left side of the object. ▪ 0x01: Wipe effect is from the bottom to the top side of the object. ▪ 0x02: Wipe effect is from the left to the right side of the object. ▪ 0x03: Wipe effect is from the top to the bottom side of the object.

Value	Description
0x0B	Zoom animation effect that uses a box pattern centered on the object instance that increases or decreases in size until the new object instance is fully shown, as illustrated in the following example:  The **animEffectDirection** value must be one of the following: ▪ 0x00: Outward direction ▪ 0x01: Inward direction
0x0C	Fly animation effect that moves the new object instance in from the specified direction to the object's on-screen location, as illustrated in the following example:  The **animEffectDirection** value must be one of the following: ▪ 0x00: Fly from the left side of the presentation slide. ▪ 0x01: Fly from the top side of the presentation slide. ▪ 0x02: Fly from the right side of the presentation slide. ▪ 0x03: Fly from the bottom side of the presentation slide. ▪ 0x04: Fly from the top-left corner of the presentation slide. ▪ 0x05: Fly from the top-right corner of the presentation slide. ▪ 0x06: Fly from the bottom-left corner of the presentation slide. ▪ 0x07: Fly from the bottom-right corner of the presentation slide. ▪ 0x08: Fly from the left edge of the shape or text. ▪ 0x09: Fly from the bottom edge of the shape or text. ▪ 0x0A: Fly from the right edge of the shape or text. ▪ 0x0B: Fly from the top edge of the shape or text. ▪ 0x0C: Crawl from the left side of the presentation slide. ▪ 0x0D: Crawl from the top side of the presentation slide. ▪ 0x0E: Crawl from the right side of the presentation slide. ▪ 0x0F: Crawl from the bottom side of the presentation slide. ▪ 0x10: The shape or text zooms in from zero size to its full size, and its center keeps unchanged. ▪ 0x11: The shape or text zooms in from half of its size to its full size, and its center remains unchanged.

Value	Description
	▪ 0x12: The shape or text zooms out from 4 times its size to its full size, and its center remains unchanged. ▪ 0x13: The shape or text zooms out from 1.5 times its size to its full size, and its center remains unchanged. ▪ 0x14: The shape or text zooms in from zero size to its full size, and its center moves from the screen center to its actual center. ▪ 0x15: The shape or text zooms out from 4 times its size to its full size, and it moves from the bottom side of the screen to its actual position. ▪ 0x16: The shape or text stretches from its center to both left and right. ▪ 0x17: The shape or text stretches from its left side to its right side. ▪ 0x18: The shape or text stretches from its top side to its bottom side. ▪ 0x19: The shape or text stretches from its right side to its left side. ▪ 0x1A: The shape or text stretches from its bottom side to its top side. ▪ 0x1B: The shape or text rotates around the vertical axis that passes through its center. ▪ 0x1C: The shape or text flies in a spiral.
0x0D	Split animation that reveals the new object instance directly on top of the previous one by wiping either horizontally or vertically from the outside in, or from the inside out, until the new object instance is fully shown, as illustrated in the following example:  The **animEffectDirection** value must be one of the following: ▪ 0x00: The split animation plays horizontally from the middle to both the top and bottom of the shape or text. ▪ 0x01: The split animation plays horizontally from the top and bottom to the middle of the shape or text. ▪ 0x02: The split animation plays vertically from the middle to both the left and right of the shape or text. ▪ 0x03: The split animation plays vertically from the left and right to the middle of the shape or text.
0x0E	Flash animation effect that displays the new object instance for a period of time and then hides the object from view, as illustrated in the following example:  **animEffectDirection** specifies the time when the flash takes place. The following table is for description purposes only. Exact time when the flash takes place is

Value	Description
	determined by the application. The **animEffectDirection** value must be one of the following: ▪ 0x00: Flash takes place after a short time. ▪ 0x01: Flash takes place after a medium time. ▪ 0x02: Flash takes place after a long time.
0x11	Diamond animation effect that uses a diamond pattern centered on the object instance that increases in size until the new object instance is fully shown, as illustrated in the following example:  **animEffectDirection** MUST be 0x00.
0x12	Plus animation effect that uses a plus pattern centered on the object instance that increases in size until the new object instance is fully shown, as illustrated in the following example:  **animEffectDirection** MUST be 0x00.
0x13	Wedge animation effect that uses two radial edges that wipe from top to bottom in opposite directions until the new object instance is fully shown, as illustrated in the following example:  **animEffectDirection** MUST be 0x00.
0x1A	Wheel animation effect that uses a set of radial edges and wipes them in the clockwise direction until the new object instance is fully shown, as illustrated in the following example:  The **animEffectDirection** value must be one of the following: ▪ 0x01: Use 1 spoke. ▪ 0x02: Use 2 spokes. ▪ 0x03: Use 3 spokes. ▪ 0x04: Use 4 spokes. ▪ 0x08: Use 8 spokes.
0x1B	Circle animation effect that uses a circle pattern centered on the object instance that

Value	Description
	increases in size until the new object instance is fully shown, as illustrated in the following example:  **animEffectDirection** MUST be 0x00.

animEffectDirection (1 byte): An **unsigned integer** that specifies the direction of the animation effect. It MUST be a value as specified by the **animEffect** field.

animAfterEffect (1 byte): An **AnimAfterEffectEnum** enumeration that specifies the behavior of the shape or text after the animation effect is finished.

textBuildSubEffect (1 byte): A **TextBuildSubEffectEnum** enumeration that specifies the behavior of text in the animation effect.

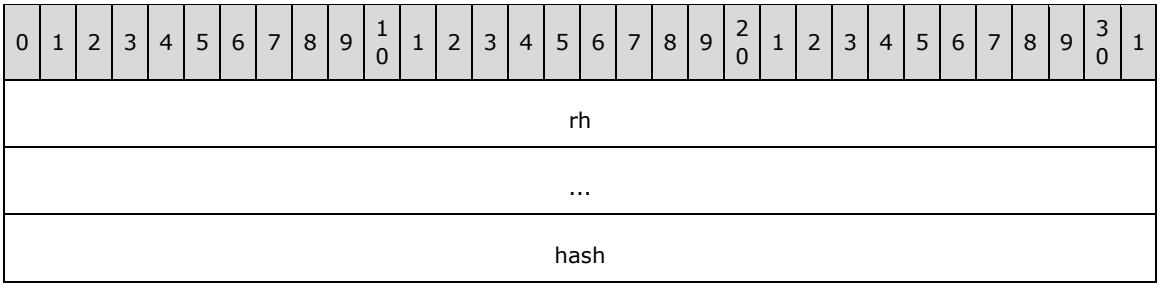
oleVerb (1 byte): An **OLEVerbEnum** enumeration that SHOULD<105> specify the OLE verb associated with this shape.

unused (2 bytes): Undefined and MUST be ignored.

2.8.3 HashCode10Atom

Referenced by: PP10SlideBinaryTagExtension

An atom record that specifies the hash code for the animation information for a slide.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_HashCodeAtom.
rh.recLen	MUST be 0x00000004.

hash (4 bytes): An **unsigned integer** that specifies a hash value for the animation information of all shapes in a slide. To calculate the hash value, a random array MUST be initialized first, as specified in the following pseudocode:

- Define **randomArray** as an array of 256 rows and 256 columns
- Initialize all elements of randomArray with 0x00000000
- Set **randomSeed** to 0x00000001
  - FOR each row of **randomArray**
    - FOR each column of **randomArray**
    - Set r0 to **randomSeed**
    - Set r1 to ((r0 * 0x000343FD + 0x00269EC3) >> 16) & 0x00007FFF
    - Set r2 to ((r1 * 0x000343FD + 0x00269EC3) >> 16) & 0x00007FFF
    - Set r3 to ((r2 * 0x000343FD + 0x00269EC3) >> 16) & 0x00007FFF
    - Set r4 to ((r3 * 0x000343FD + 0x00269EC3) >> 16) & 0x00007FFF
    - Set **randomSeed** to r4
    - Set r1 to (r1 % 0x00000100)
    - Set r2 to (r2 % 0x00000100) << 8
    - Set r3 to (r3 % 0x00000100) << 16
    - Set r4 to (r4 % 0x00000100) << 24
    - Set **randomArray** position (row, column) to r4 | r3 | r2 | r1
    - END FOR
  - END FOR

Then, the random array can be used to calculate hash values for all slides, as specified in the following pseudocode:

- Set hash to 0x00000000
  - FOR each shape in the slide
  - Define **animInfoAtom** as **AnimationInfoAtom**
  - Initialize all bytes of **animInfoAtom** with zero
    - IF **AnimationInfoAtom** exists in the shape THEN
    - Read the animation information into **animInfoAtom**
    - END IF
  - Set **shapeId** to identifier of the shape
    - FOR each byte in **animInfoAtom**
    - Set **byteIndex** to the index of the byte in **animInfoAtom**
    - Set **rowIndex** to (shapeId * (**byteIndex** + 1)) % 256
    - Set **hash** to **hash** ^ **randomArray**[**rowIndex**][**byte**]

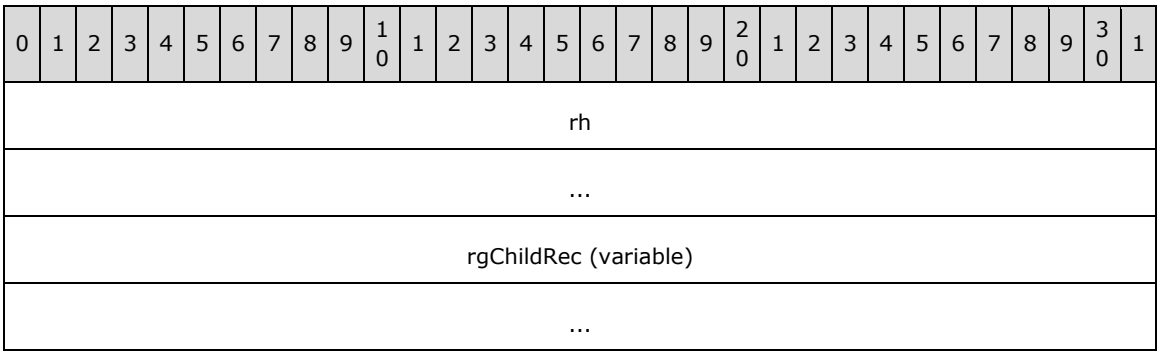
- END FOR

- END FOR

2.8.4 BuildListContainer

Referenced by: PP10SlideBinaryTagExtension

A container record that specifies all animation builds for a slide.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_BuildList.

rgChildRec (variable): An array of BuildListSubContainer records that specifies all builds for a slide. Each item specifies build information for a shape.

2.8.5 BuildListSubContainer

Referenced by: BuildListContainer

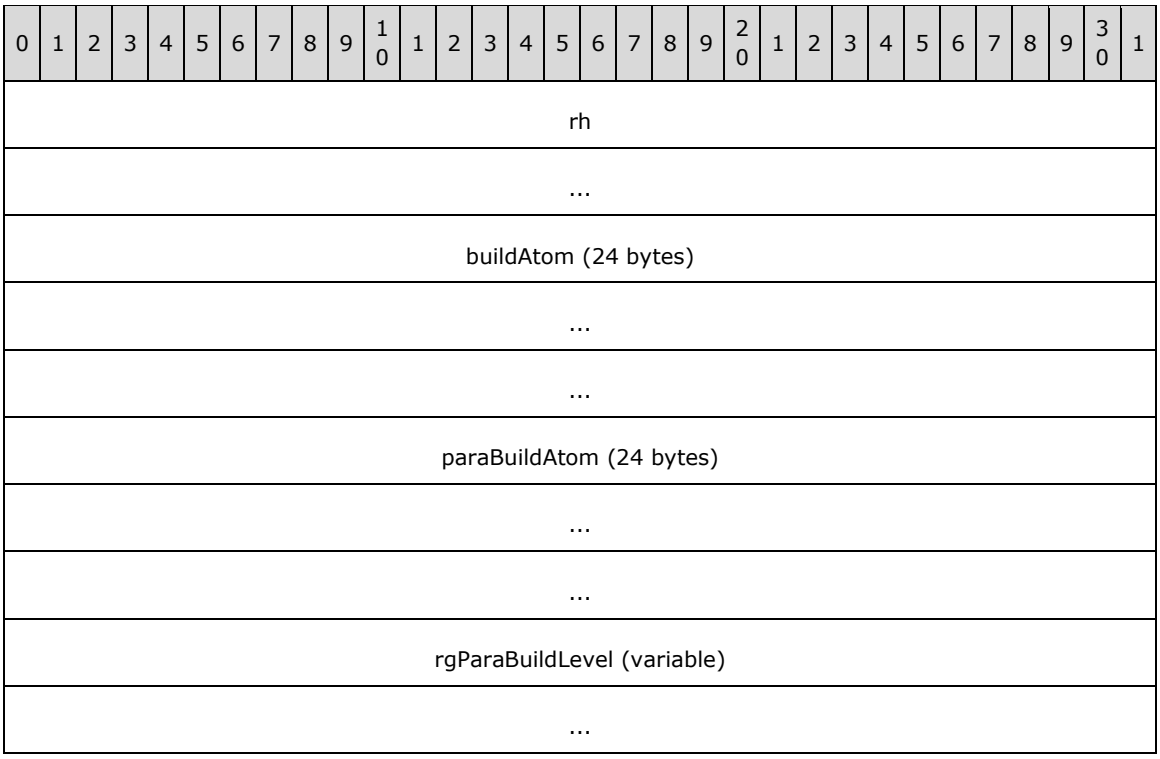
A variable type record whose type and meaning are dictated by the value of **rh.recType**, as specified in the following table.

Value	Meaning
RT_ParaBuild	A ParaBuildContainer record that specifies text build information.
RT_ChartBuild	A ChartBuildContainer record that specifies chart build information.
RT_DiagramBuild	A DiagramBuildContainer record that specifies diagram build information.

2.8.6 ParaBuildContainer

Referenced by: BuildListSubContainer

A container record that specifies the build information for text paragraphs in a shape.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_ParaBuild.

buildAtom (24 bytes): A BuildAtom record that specifies the information for the build.

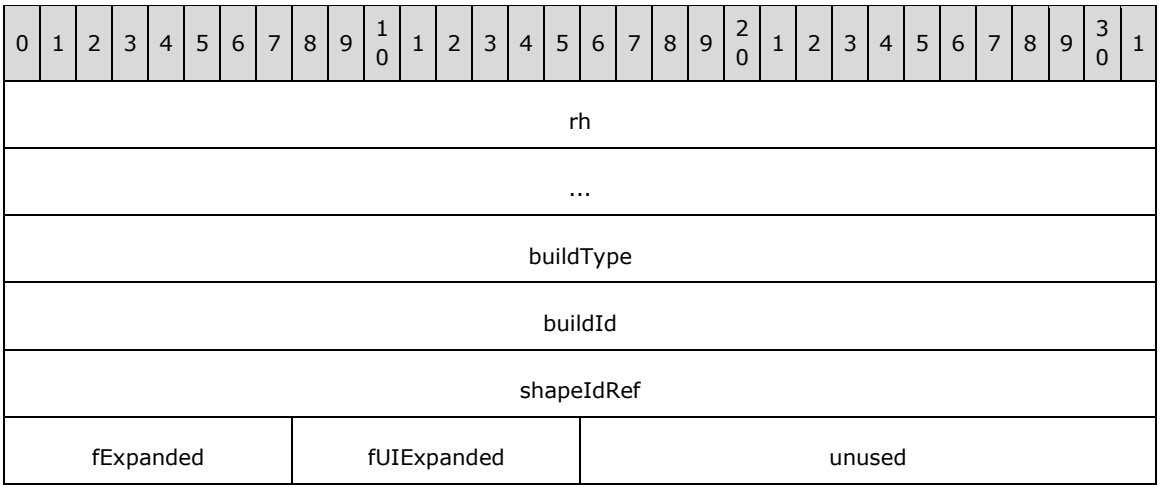
paraBuildAtom (24 bytes): A ParaBuildAtom record that specifies the information for the paragraph build.

rgParaBuildLevel (variable): An array of ParaBuildLevel that specifies the template effects for the text. If **paraBuildAtom**.**paraBuild** is TLPB_AsAWhole, **rgParaBuildLevel** MUST contain one and only one ParaBuildLevel that specifies the template effects for the text. Otherwise, **rgParaBuildLevel** MUST contain the same number of ParaBuildLevel items as the number of paragraph levels in the shape, ordered from level 1 to the biggest level. Each ParaBuildLevel item in the array specifies the template effects for a paragraph level in the text.

2.8.7 BuildAtom

Referenced by: ChartBuildContainer, DiagramBuildContainer, ParaBuildContainer

An atom record that specifies the build information for a chart, diagram or text.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_BuildAtom.
rh.recLen	MUST be 0x00000010.

buildType (4 bytes): A **BuildTypeEnum** enumeration that specifies the build type.

buildId (4 bytes): An unsigned integer that specifies the **build identifier**. The combination of **buildId** and **shapeIdRef** MUST be unique for all builds in the slide.

shapeIdRef (4 bytes): An unsigned integer that specifies the target shape that this build is applied to.

fExpanded (1 byte): A **bool1** (section 2.2.2) that specifies whether this build has been expanded into time nodes.

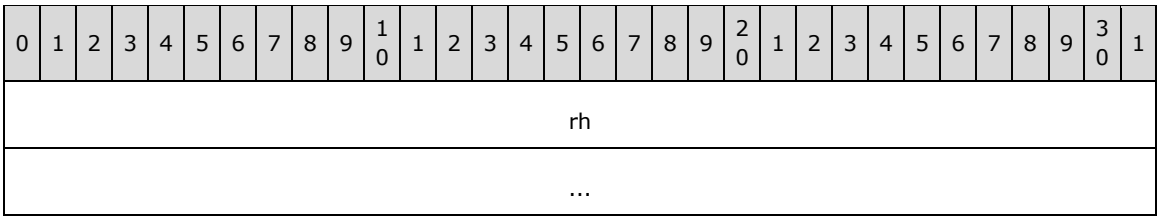
fUIExpanded (1 byte): A **bool1** that specifies whether this build is shown as expanded in the user interface.

unused (2 bytes): Undefined and MUST be ignored.

2.8.8 ParaBuildAtom

Referenced by: ParaBuildContainer

An atom record that specifies the information for a paragraph build.



paraBuild			
buildLevel			
fAnimBackground	fReverse	fUserSetAnimBackground	fAutomatic
delayTime			

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x1.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_ParaBuildAtom.
rh.recLen	MUST be 0x00000010.

paraBuild (4 bytes): A **ParaBuildEnum** enumeration that specifies the paragraph build type for text in the shape.

buildLevel (4 bytes): An unsigned integer that specifies which paragraph level the paragraph build applies to. Paragraphs in level 1 to level **buildLevel** expand individually, and paragraphs in other levels animate at the same time as level **buildLevel**. If **paraBuild** is not TLPB_BuildByNthLevel, **buildLevel** MUST be ignored.

fAnimBackground (1 byte): A **bool1** (section 2.2.2) that specifies whether the background animates. If there is an attached shape or a background on the text box or text placeholder shape, **fAnimBackground** is valid; otherwise, it MUST be ignored. It MUST be a value from the following table.

Value	Meaning
0x00	The background is not animated.
0x01	The background is animated.

fReverse (1 byte): A **bool1** that specifies whether the animation plays in reverse order. It MUST be a value from the following table:

Value	Meaning
0x00	The animation does not play in reverse order.
0x01	The animation plays in reverse order.

fUserSetAnimBackground (1 byte): A **bool1** that specifies whether **fAnimBackground** has been toggled by any users. It MUST be a value from the following table.

Value	Meaning
0x00	**fAnimBackground** has never been toggled.
0x01	**fAnimBackground** has been toggled.

fAutomatic (1 byte): A **bool1** that specifies whether to automatically proceed to the next build step after the current build step is finished. It MUST be a value from the following table.

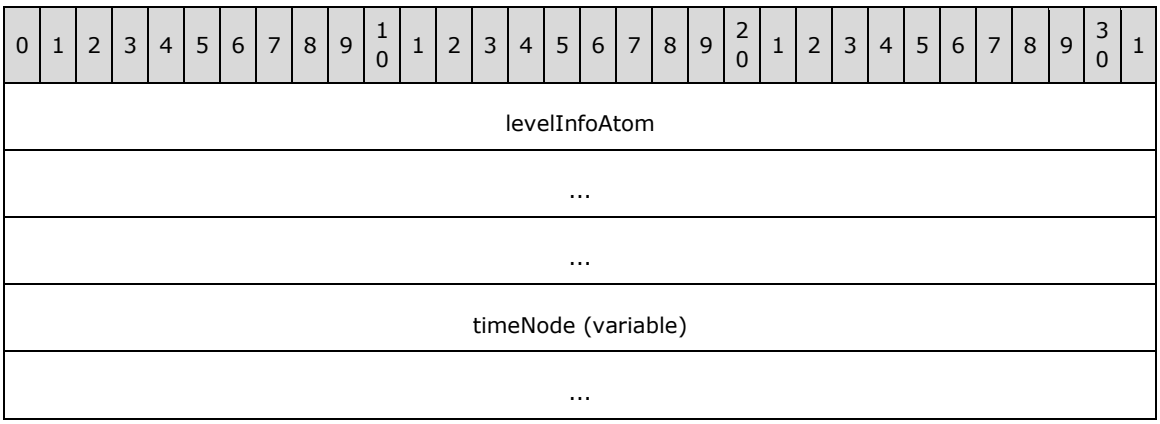
Value	Meaning
0x00	Do not automatically proceed to the next build step.
0x01	Automatically proceed to the next build step.

delayTime (4 bytes): An unsigned integer that specifies the waiting time, in milliseconds, from the time when the current build step is finished to the time when the next build step starts. It MUST be ignored if **fAutomatic** is 0x00.

2.8.9 ParaBuildLevel

Referenced by: ParaBuildContainer

A structure that specifies information about the build step for a specific paragraph level.



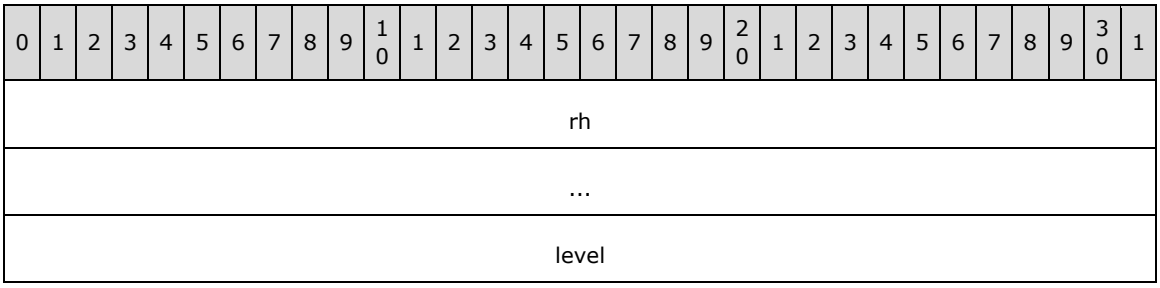
levelInfoAtom (12 bytes): A LevelInfoAtom record that specifies the paragraph level.

timeNode (variable): An **ExtTimeNodeContainer** record (section 2.8.15) that specifies all time nodes for the paragraph level specified by the **levelInfoAtom**.

2.8.10 LevelInfoAtom

Referenced by: ParaBuildLevel

An atom record that specifies the level for a paragraph.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

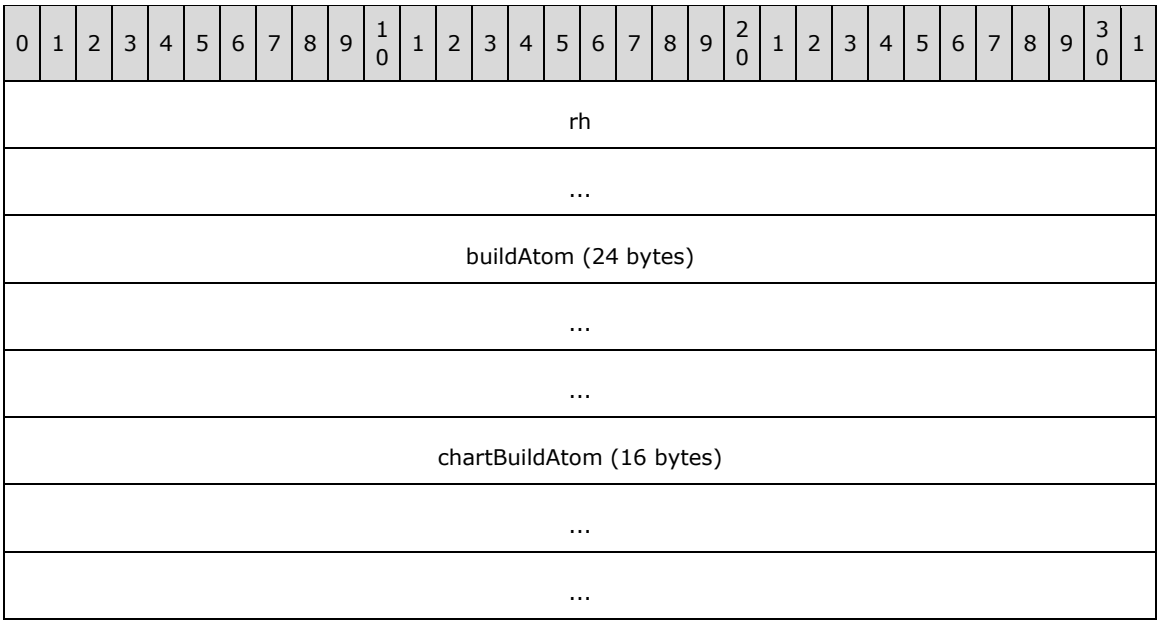
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_LevelInfoAtom.
rh.recLen	MUST be 0x00000004

level (4 bytes): An **unsigned integer** that specifies the paragraph level. It MUST be less than or equal to 0x00000009. It SHOULD<106> be less than or equal to 0x00000005.

2.8.11 ChartBuildContainer

Referenced by: BuildListSubContainer

A container record that specifies the build information for a chart.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ChartBuild.
rh.recLen	MUST be 0x00000028.

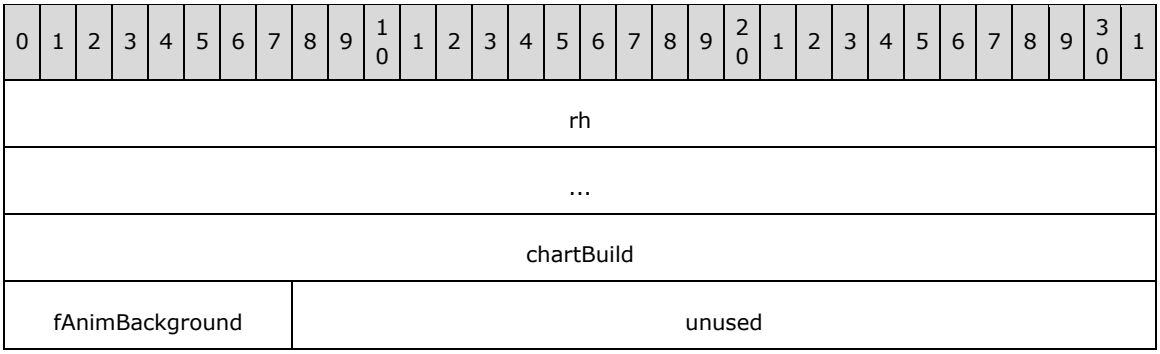
buildAtom (24 bytes): A BuildAtom record that specifies the information for the build.

chartBuildAtom (16 bytes): A ChartBuildAtom record that specifies the information for the chart build.

2.8.12 ChartBuildAtom

Referenced by: ChartBuildContainer

An atom record that specifies the information for a chart build.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ChartBuildAtom.
rh.recLen	MUST be 0x00000008.

chartBuild (4 bytes): A **ChartBuildEnum** enumeration that specifies the chart build type.

fAnimBackground (1 byte): A **bool1** (section 2.2.2) that specifies whether the background of the chart animates separately. If **chartBuild** is TLCB_AsOneObject, **fAnimBackground** MUST be ignored. It MUST be a value from the following table.

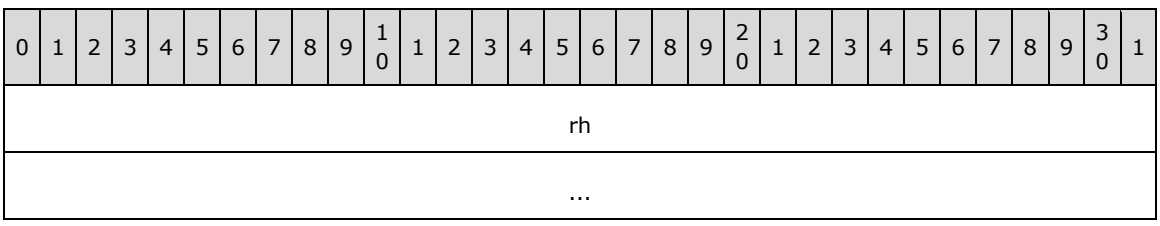
Value	Meaning
0x00	Do not animate the background.
0x01	Animate the background.

unused (3 bytes): Undefined and MUST be ignored.

2.8.13 DiagramBuildContainer

Referenced by: BuildListSubContainer

A container record that specifies the build information for a diagram.



buildAtom (24 bytes)
...
...
diagramBuildAtom
...
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_DiagramBuild.
rh.recLen	MUST be 0x00000024.

buildAtom (24 bytes): A BuildAtom record that specifies the information for the build.

diagramBuildAtom (12 bytes): A DiagramBuildAtom record that specifies the information for the diagram build.

2.8.14 DiagramBuildAtom

Referenced by: DiagramBuildContainer

An atom record that specifies the information for a diagram build.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
diagramBuild																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_DiagramBuildAtom.
rh.recLen	MUST be 0x00000004.

diagramBuild (4 bytes): A **DiagramBuildEnum** enumeration that specifies the diagram build type.

2.8.15 ExtTimeNodeContainer

Referenced by: ParaBuildLevel, PP10SlideBinaryTagExtension

A container record that specifies a time node. This time node is used to store all information necessary to cause a time-based or an action-based effect to occur during a slide show. Each time node effect has a corresponding object to which the effect applies.

At most one of the following fields MUST exist: **timeAnimateBehavior**, **timeColorBehavior**, **timeEffectBehavior**, **timeMotionBehavior**, **timeRotationBehavior**, **timeScaleBehavior**, **timeSetBehavior**, or **timeCommandBehavior**.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
timeNodeAtom (40 bytes)																															
...																															
...																															
timePropertyList (variable)																															
...																															
timeAnimateBehavior (variable)																															
...																															
timeColorBehavior (variable)																															
...																															
timeEffectBehavior (variable)																															
...																															
timeMotionBehavior (variable)																															
...																															
timeRotationBehavior (variable)																															
...																															
timeScaleBehavior (variable)																															

...
timeSetBehavior (variable)
...
timeCommandBehavior (variable)
...
clientVisualElement (variable)
...
timeIterateDataAtom (28 bytes, optional)
...
...
timeSequenceDataAtom (28 bytes, optional)
...
...
rgBeginTimeCondition (variable)
...
rgEndTimeCondition (variable)
...
timeEndSyncTimeCondition (variable)
...
rgTimeModifierAtom (variable)
...
rgSubEffect (variable)
...
rgExtTimeNodeChildren (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Subfields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be an RT_TimeExtTimeNodeContainer.

timeNodeAtom (40 bytes): A TimeNodeAtom record that specifies time-based attributes of this time node.

timePropertyList (variable): An optional **TimePropertyList4TimeNodeContainer** record (section 2.8.18) that specifies a list of attributes of the time node.

timeAnimateBehavior (variable): An optional **TimeAnimateBehaviorContainer** record (section 2.8.29) that specifies a generic animation behavior that can animate any property of an object. It MUST exist only if **timeNodeAtom.type** is TL_TNT_Behavior.

timeColorBehavior (variable): An optional **TimeColorBehaviorContainer** record (section 2.8.52) that specifies a color animation behavior that changes the color of an object. It MUST only if **timeNodeAtom.type** is TL_TNT_Behavior.

timeEffectBehavior (variable): An optional **TimeEffectBehaviorContainer** record (section 2.8.61) that specifies an effect-animation behavior that transforms the image of an object. It MUST exist only if **timeNodeAtom.type** is TL_TNT_Behavior.

timeMotionBehavior (variable): An optional **TimeMotionBehaviorContainer** record (section 2.8.63) that specifies a motion-animation behavior that moves a positioned object along a path. It MUST exist only if **timeNodeAtom.type** is TL_TNT_Behavior.

timeRotationBehavior (variable): An optional **TimeRotationBehaviorContainer** record (section 2.8.65) that specifies a rotation animation behavior that rotates an object. It MUST exist only if **timeNodeAtom.type** is TL_TNT_Behavior.

timeScaleBehavior (variable): An optional **TimeScaleBehaviorContainer** record (section 2.8.67) that specifies a scale-animation behavior that changes the size of an object. It MUST exist only if **timeNodeAtom.type** is TL_TNT_Behavior.

timeSetBehavior (variable): An optional **TimeSetBehaviorContainer** record (section 2.8.69) that specifies a set-animation behavior that assigns a value to a property of an object. It MUST exist if **timeNodeAtom.type** is TL_TNT_Behavior.

timeCommandBehavior (variable): An optional **TimeCommandBehaviorContainer** record (section 2.8.71) that specifies a command-animation behavior that performs a command as an animation. It MUST exist only if **timeNodeAtom.type** field is TL_TNT_Behavior.

clientVisualElement (variable): An optional **ClientVisualElementContainer** record (section 2.8.44) that specifies a media file to be played. It MUST exist only if **timeNodeAtom.type** is TL_TNT_Media.

timeIterateDataAtom (28 bytes): An optional TimeIterateDataAtom record that specifies how an animation is applied to the subelements of a target object for a repeated effect.

timeSequenceDataAtom (28 bytes): An optional TimeSequenceDataAtom record that specifies sequencing information for the child nodes of this time node. It MUST exist only if **timeNodeAtom.type** is TL_TNT_Sequential.

rgBeginTimeCondition (variable): An optional array of **TimeConditionContainer** record (section 2.8.75) that specifies the **time conditions** that MUST be used in one of the following situations:

- When the **rh.recInstance** field of a **TimeConditionContainer** record is TL_CT_Begin, any of these time conditions determines when this time node will be activated.
- When the **rh.recInstance** field of a **TimeConditionContainer** record is TL_CT_Next and the **timeNodeAtom.type** field is TL_TNT_Sequential, any of these time conditions determines when the next child time node will be activated.

  The array continues while the **rh.recType** field of the **TimeConditionContainer** record is equal to RT_TimeConditionContainer and one of the two aforementioned situations applies.

rgEndTimeCondition (variable): An optional array of **TimeConditionContainer** records (section 2.8.75) that specifies the time conditions that MUST be utilized in one of the following situations:

- When the **rh.recInstance** field of a **TimeConditionContainer** record is TL_CT_End, any of these time conditions determines when this time node will be deactivated.
- When the **rh.recInstance** field of a **TimeConditionContainer** record is TL_CT_Previous and the **timeNodeAtom.type** field is TL_TNT_Sequential, any of these time conditions determines when the next child time node will be deactivated.

  The array continues while the **rh.recType** field of the **TimeConditionContainer** record is equal to RT_TimeConditionContainer and one of the two aforementioned situations is applies.

timeEndSyncTimeCondition (variable): An optional **TimeConditionContainer** record (section 2.8.75) that specifies how to synchronize the stopping of the child nodes of this time node. The **timeEndSyncTimeCondition.rh.recInstance** sub-field MUST be TL_CT_EndSync.

rgTimeModifierAtom (variable): An optional array of TimeModifierAtom records that specifies the modification records that store the type of data to be modified and the new data value. The array continues while the **rh.recType** field of the TimeModifierAtom record is equal to RT_TimeModifier.

rgSubEffect (variable): An optional array of SubEffectContainer that specifies the subordinate time nodes whose start time depends on the relation to this time node. The relationship is specified in the TimeMasterRelType record contained in the **timePropertyList** field of the SubEffectContainer record. The array continues while the **rh.recType** field of the SubEffectContainer record is equal to RT_TimeSubEffectContainer.

rgExtTimeNodeChildren (variable): An array of **ExtTimeNodeContainer** that specifies the child time nodes of this time node.

2.8.16 SubEffectContainer

A container record that specifies a subordinate time node whose start time depends on the relation to its master time node.

At most one of the following fields MUST exist: **timeColorBehavior**, **timeSetBehavior**, or **timeCommandBehavior**.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															

timeNodeAtom (40 bytes)
...
...
timePropertyList (variable)
...
timeColorBehavior (variable)
...
timeSetBehavior (variable)
...
timeCommandBehavior (variable)
...
clientVisualElement (variable)
...
rgBeginTimeCondition (variable)
...
rgEndTimeCondition (variable)
...
rgTimeModifierAtom (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Subfields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be an RT_TimeSubEffectContainer.

timeNodeAtom (40 bytes): A TimeNodeAtom record that specifies some attributes of this subordinate time node. The **timeNodeAtom.type** subfield MUST be TL_TNT_Behavior or TL_TNT_Media.

timePropertyList (variable): An optional **TimePropertyList4TimeNodeContainer** record (section 2.8.18) that specifies a list of attributes of the subordinate time node.

timeColorBehavior (variable): An optional **TimeColorBehaviorContainer** record (section 2.8.52) that specifies a color animation behavior that changes the color of an object. It MUST exist only if **timeNodeAtom.type** is TL_TNT_Behavior.

timeSetBehavior (variable): An optional **TimeSetBehaviorContainer** record (section 2.8.69) that specifies a set animation behavior that assigns a value to a property of an object. It MUST exist only if **timeNodeAtom.type** is TL_TNT_Behavior.

timeCommandBehavior (variable): An optional **TimeCommandBehaviorContainer** record (section 2.8.71) that specifies a command-animation behavior that performs a command as an animation. It MUST exist if and only if the **timeCommandBehavior.rh.recType** field is RT_TimeCommandBehaviorContainer and the **timeNodeAtom.type** field is TL_TNT_Behavior.

clientVisualElement (variable): An optional **ClientVisualElementContainer** record (section 2.8.44) that specifies a media to be played. It MUST exist only if **timeNodeAtom.type** is TL_TNT_Media.

rgBeginTimeCondition (variable): An optional array of **TimeConditionContainer** records (section 2.8.75) that specifies the time conditions. It MUST be used when the **rh.recInstance** field of a **TimeConditionContainer** record is TL_CT_Begin. Any of these time conditions determine when this subordinate time node will be activated.

The array continues while the **rh.recType** field of the **TimeConditionContainer** record is equal to RT_TimeConditionContainer and the situation in the previous paragraph applies.

rgEndTimeCondition (variable): An optional array of **TimeConditionContainer** records that specifies the time conditions. It MUST be used when the **rh.recInstance** field of a **TimeConditionContainer** item is TL_CT_End. Any of these time conditions determines when this subordinate time node will be deactivated.

The array continues while the **rh.recType** field of the **TimeConditionContainer** record is equal to RT_TimeConditionContainer and the previous situation applies.

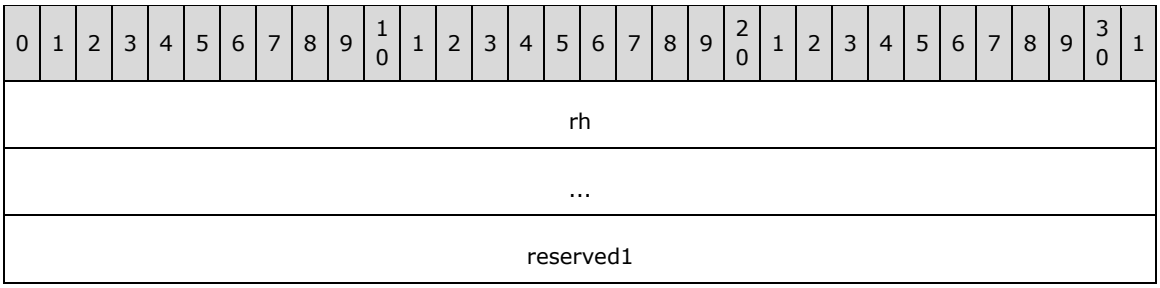
rgTimeModifierAtom (variable): An optional array of TimeModifierAtom records that specifies the modification records that store the type of data to be modified and the new data value. The array continues while the **rh.recType** field of the TimeModifierAtom record is equal to RT_TimeModifier.

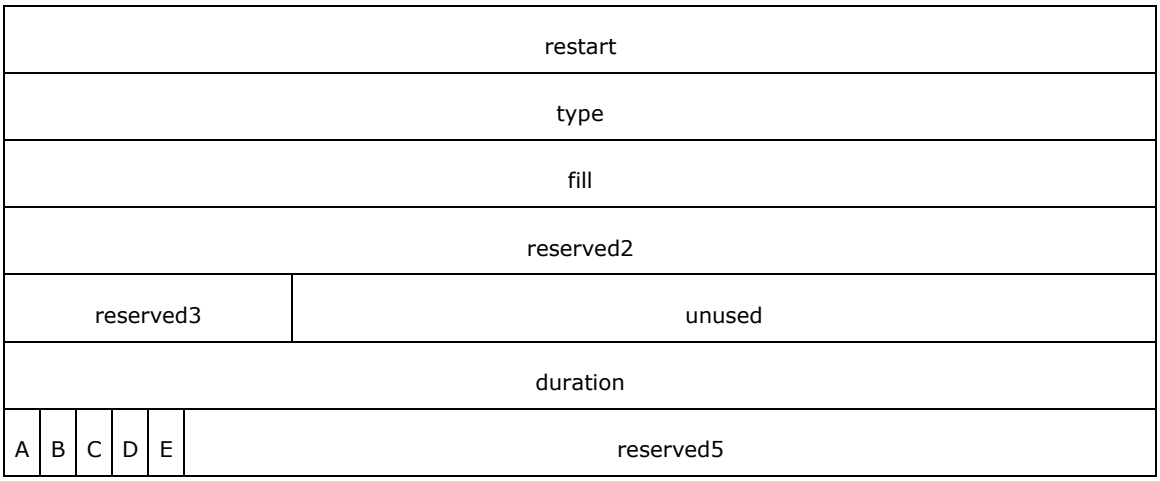
2.8.17 TimeNodeAtom

Referenced by: ExtTimeNodeContainer, SubEffectContainer

An atom record that specifies the attributes of a time node.

Let the *corresponding time node* be specified by the **ExtTimeNodeContainer** record (section 2.8.15) or the SubEffectContainer record that contains this **TimeNodeAtom** record.





rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeNode.
rh.recLen	MUST be 0x00000020.

reserved1 (4 bytes): MUST be zero, and MUST be ignored.

restart (4 bytes): An **unsigned integer** that specifies how the *corresponding time node* restarts when it completes its action. It MUST be ignored if **fRestartProperty** is **FALSE** and a value of 0x00000000 MUST be used instead. It MUST be a value from the following table:

Value	Meaning
0x00000000	Does not restart.
0x00000001	Can restart at any time.
0x00000002	Can restart when the *corresponding time node* is not active.
0x00000003	Same as 0x00000000.

type (4 bytes): A **TimeNodeTypeEnum** enumeration that specifies the type of the *corresponding time node*. It MUST be ignored if **fGroupingTypeProperty** is **FALSE** and a value of TL_TNT_Parallel MUST be used instead.

fill (4 bytes): An **unsigned integer** that specifies the state of the target object's properties when the animation ends. It MUST be ignored if **fFillProperty** is **FALSE** and a value of 0x00000000 MUST be used instead. It MUST be a value from the following table.

Value	Meaning
0x00000000	The properties remain at their ending values while the parent time node is still running or holding. After which, the properties reset to their original values.
0x00000001	The properties reset to their original values after

Value	Meaning
	the time node becomes inactive.
0x00000002	The properties remain at their ending values while the parent time node is still running or holding, or until another sibling time node is started under a sequential time node as specified in the **type** field. After which, the properties reset to their original values.
0x00000003	Same as 0x00000000.
0x00000004	Same as 0x00000001.

reserved2 (4 bytes): MUST be zero, and MUST be ignored.

reserved3 (1 byte): MUST be zero, and MUST be ignored.

unused (3 bytes): Undefined and MUST be ignored.

duration (4 bytes): A signed integer that specifies the duration of the *corresponding time node* in milliseconds. It MUST be ignored if **fDurationProperty** is **FALSE** and a value of 0x00000000 MUST be used instead.

A - fFillProperty (1 bit): A bit that specifies whether **fill** was explicitly set by a user interface action.

B - fRestartProperty (1 bit): A bit that specifies whether **restart** was explicitly set by a user interface action.

C - reserved4 (1 bit): MUST be zero, and MUST be ignored.

D - fGroupingTypeProperty (1 bit): A bit that specifies whether **type** was explicitly set by a user interface action.

E - fDurationProperty (1 bit): A bit that specifies whether **duration** was explicitly set by a user interface action.

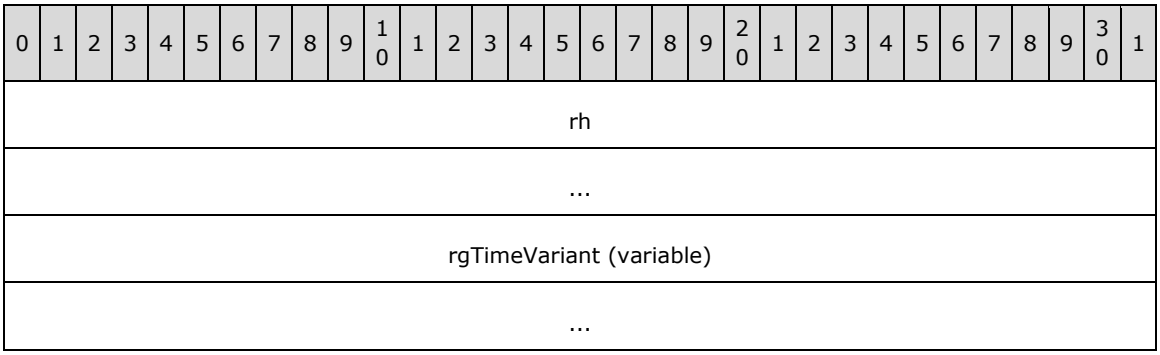
reserved5 (27 bits): MUST be zero, and MUST be ignored.

2.8.18 TimePropertyList4TimeNodeContainer

Referenced by: ExtTimeNodeContainer, SubEffectContainer

A container record that specifies a list of attributes for a time node.

Let the *corresponding time node* be specified by the **ExtTimeNodeContainer** record (section 2.8.15) or the **SubEffectContainer** record (section 2.8.16) that contains this **TimePropertyList4TimeNodeContainer** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Subfields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_TimePropertyList.

rgTimeVariant (variable): An array of TimeVariant2.8.19TimeNode records that specifies the list of attributes for the *corresponding time node*. The size, in bytes, of the array is specified by **rh.recLen**. Each **TimePropertyID2.13.38TimeNode** enumeration value MUST NOT occur more than once as a value of the **rh.recInstance** field in the array.

If the TL_TPID_AfterEffect value does not occur, a TimeVariantBool record in which the **boolValue** field is set to 0x00 SHOULD be used.

If the TL_TPID_Display value does not occur, a TimeDisplayType record in which the **displayType** field is set to 0x00000000 SHOULD be used.

If the *corresponding time node* is an **ExtTimeNodeContainer** record (section 2.8.15), the following values MUST NOT occur: TL_TPID_MasterPos and TimeSubType.

If the *corresponding time node* is a **SubEffectContainer** record, the following values MUST NOT occur: TL_TPID_EffectID, TL_TPID_EffectDir, TL_TPID_EffectType, TL_TPID_SlideCount, TL_TPID_TimeFilter, TL_TPID_EventFilter, TL_TPID_HideWhenStopped, TL_TPID_GroupID, TL_TPID_EffectNodeType, and TL_TPID_ZoomToFullScreen.

2.8.19 TimeVariant4TimeNode

Referenced by: TimePropertyList4TimeNodeContainer

A variable type record that specifies an attribute of a time node and whose type and meaning are specified by the value of the **rh.recInstance** field, as specified in the following table.

Value	Meaning
TL_TPID_Display	A TimeDisplayType record that specifies the visibility of the time node in the user interface.
TL_TPID_MasterPos	A TimeMasterRelType record that specifies the relationship of a subordinate time node to its master time node.
TL_TPID_SubType	A TimeSubType record that specifies the type of subordinate time node.
TL_TPID_EffectID	A TimeEffectID record that specifies an identifier of an animation effect.
TL_TPID_EffectDir	A TimeVariantInt record that specifies the direction or other attributes of an animation effect. The values for each animation effect are specified in the TimeEffectID record.
TL_TPID_EffectType	A TimeEffectType record that specifies the type of animation effect.
TL_TPID_AfterEffect	A TimeVariantBool record that specifies whether the time node is an after effect.
TL_TPID_SlideCount	A TimeVariantInt record that specifies the number of slides that a media will play across.
TL_TPID_TimeFilter	A TimeNodeTimeFilter record that specifies a time filter that transforms a given

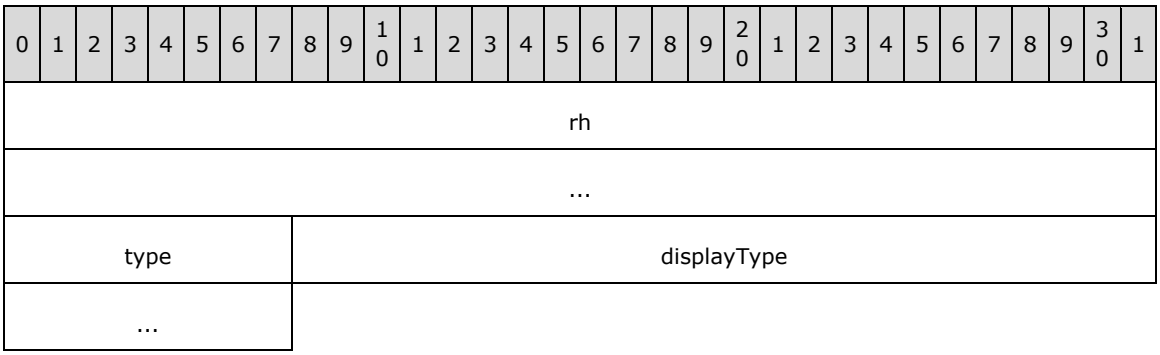
Value	Meaning
	time to the local time of a time node.
TL_TPID_EventFilter	A TimeEventFilter structure that specifies an event filter for a time node.
TL_TPID_HideWhenStopped	A TimeVariantBool structure that specifies whether to display the media when it is stopped.
TL_TPID_GroupID	A TimeGroupID structure that specifies a reference to a build identifier of an animation effect.
TL_TPID_EffectNodeType	A TimeEffectNodeType structure that specifies the role of a time node in the timing structure.
TL_TPID_PlaceholderNode	A TimeVariantBool structure that specifies whether the time node is a placeholder node that is not played during a slide show.
TL_TPID_MediaVolume	A TimeVariantFloat structure that specifies the volume of a media. The **floatValue** sub-field MUST be greater than or equal to 0 and less than or equal to 100000.
TL_TPID_MediaMute	A TimeVariantBool structure that specifies whether a media is muted.
TL_TPID_ZoomToFullScreen	A TimeVariantBool structure that specifies whether to zoom a media to full screen when it plays.

2.8.20 TimeDisplayType

Referenced by: TimeVariant4TimeNode

An atom record that specifies whether a time node is visible in the user interface.

Let the *corresponding time node* be as specified in the **TimePropertyList4TimeNodeContainer** record (section 2.8.18) that contains this **TimeDisplayType** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be an RT_TimeVariant.
rh.recLen	MUST be 0x00000005.

type (1 byte): A **TimeVariantTypeEnum** enumeration that specifies the data type of this record. It MUST be TL_TVT_Int.

displayType (4 bytes): A signed integer that specifies whether the *corresponding time node* is displayed in the user interface. It MUST be a value from the following table:

Value	Meaning
0x00000000	The *corresponding time node* is visible.
0x00000001	The *corresponding time node* is hidden.

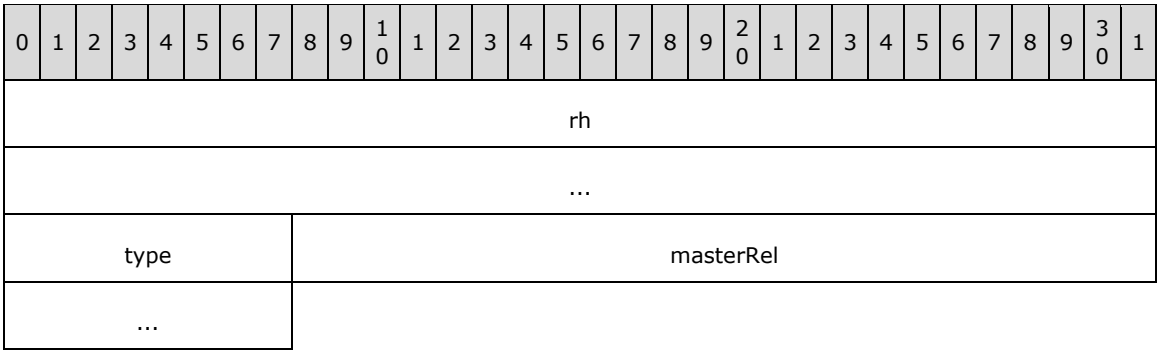
2.8.21 TimeMasterRelType

Referenced by: *TimeVariant4TimeNode*

An atom record that specifies how a subordinate time node plays back relative to its master time node.

Let the *corresponding subordinate time node* be specified by the **SubEffectContainer** record (section 2.8.16) that contains this **TimeMasterRelType** record.

Let the *corresponding master time node* be specified by the **ExtTimeNodeContainer** record (section 2.8.15) that contains the *corresponding subordinate time node*.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be RT_TimeVariant.
rh.recLen	MUST be 0x00000005.

type (1 byte): A **TimeVariantTypeEnum** enumeration that specifies the data type of this record. It MUST be TL_TVT_Int.

masterRel (4 bytes): A **signed integer** that specifies how the *corresponding subordinate time node* plays back relative to the *corresponding master time node*. It MUST be a value from the following table.

Value	Meaning
0x00000000	Do not start the *corresponding subordinate time node*.
0x00000002	Start the *corresponding subordinate time node* when the *corresponding master time node* starts.

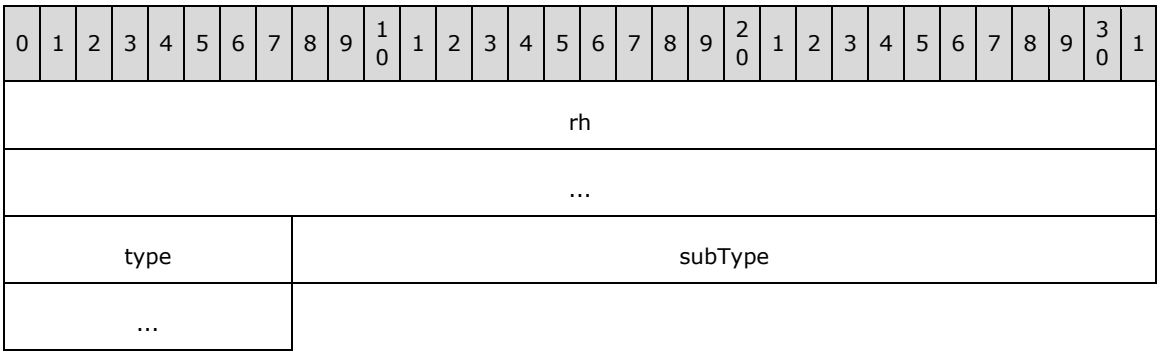
2.8.22 TimeSubType

Referenced by: TimeVariant4TimeNode

An atom record that specifies the type of subordinate time node.

Let the *corresponding subordinate time node* be specified by the **SubEffectContainer** record (section 2.8.16) that contains this **TimeSubType** record.

Let the *corresponding master time node* be specified by the **ExtTimeNodeContainer** record (section 2.8.15) that contains the *corresponding subordinate time node*.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Subfields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be an RT_TimeVariant.
rh.recLen	MUST be 0x00000005.

type (1 byte): A **TimeVariantTypeEnum** enumeration that specifies the data type of this record. It MUST be TL_TVT_Int.

subType (4 bytes): A signed integer that specifies the type of the *corresponding subordinate time node*. It MUST be 0x00000001, specifying that the corresponding *subordinate time node* position is relative to the *corresponding master time node*.

2.8.23 TimeEffectID

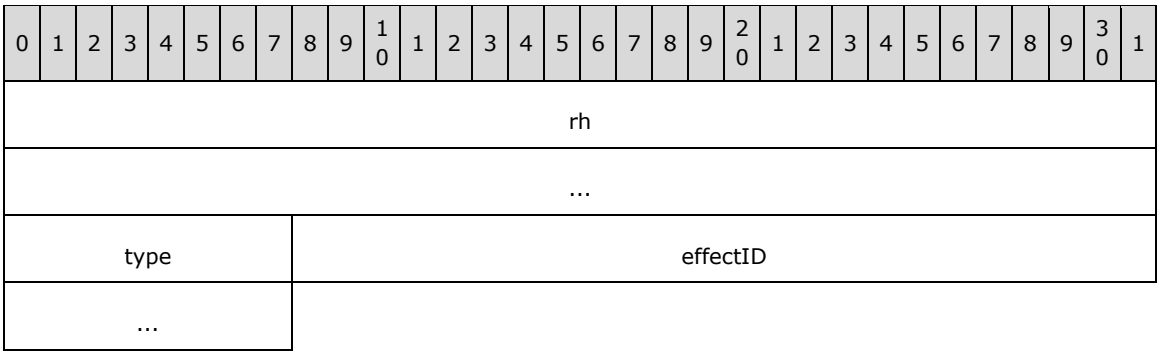
Referenced by: TimeVariant4TimeNode

An atom record that specifies the identifier of an animation effect.

Let the *corresponding time node* be as specified in the **TimePropertyList4TimeNodeContainer** record (section 2.8.18) that contains this **TimeEffectID** record.

Let the *corresponding effect type* be specified by the TimeEffectType record contained in the **TimePropertyList4TimeNodeContainer** record that contains this **TimeEffectID** record.

Let the *corresponding effect direction* be specified by the **intValue** field of the TimeVariantInt record contained in the **TimePropertyList4TimeNodeContainer** record such that the **rh.recInstance** field of the TimeVariantInt record is equal to TL_TPID_EffectDir.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be RT_TimeVariant.
rh.recLen	MUST be 0x00000005.

type (1 byte): A **TimeVariantTypeEnum** enumeration that specifies the data type of this record. It MUST be TL_TVT_Int.

effectID (4 bytes): A signed integer that specifies the identifier of the animation effect of the *corresponding time node*. When the *corresponding effect type* is an entrance or an exit effect, this field MUST be a value from the following table:

Value	Meaning
0x00000000	Custom. The *corresponding effect direction* MUST be ignored.
0x00000001	Appear. The *corresponding effect direction* MUST be ignored.
0x00000002	Fly in. The *corresponding effect direction* MUST be one of the following values: ▪ 0x00000002: Right ▪ 0x00000008: Left ▪ 0x00000001: Top ▪ 0x00000004: Bottom ▪ 0x00000009: Top left ▪ 0x00000003: Top right ▪ 0x00000006: Bottom right ▪ 0x0000000C: Bottom left

Value	Meaning
0x00000003	Blinds. The *corresponding effect direction* MUST be one of the following values: ▪ 0x0000000A: Horizontal ▪ 0x00000005: Vertical
0x00000004	Box. The *corresponding effect direction* MUST be one of the following values: ▪ 0x00000010: In ▪ 0x00000020: Out
0x00000005	Check board. The *corresponding effect direction* MUST be one of the following values: ▪ 0x00000005: Vertical ▪ 0x0000000A: Across
0x00000006	Circle. The *corresponding effect direction* MUST be one of the following values: ▪ 0x00000010: In ▪ 0x00000020: Out
0x00000007	Crawl. The *corresponding effect direction* MUST be one of the following values: ▪ 0x00000002: Right ▪ 0x00000008: Left ▪ 0x00000001: Top ▪ 0x00000004: Bottom ▪ 0x00000009: Top left ▪ 0x00000003: Top right ▪ 0x00000006: Bottom right

Value	Meaning
	▪ 0x0000000C: Bottom left
0x00000008	Diamond. The *corresponding effect direction* MUST be one of the following values: ▪ 0x00000010: In ▪ 0x00000020: Out
0x00000009	Dissolve. The *corresponding effect direction* MUST be ignored.
0x0000000A	Fade. The *corresponding effect direction* MUST be ignored.
0x0000000B	Flash once. The *corresponding effect direction* MUST be ignored.
0x0000000C	Peek. The *corresponding effect direction* MUST be one of the following values: ▪ 0x00000002: Right ▪ 0x00000008: Left ▪ 0x00000001: Top ▪ 0x00000004: Bottom
0x0000000D	Plus. The *corresponding effect direction* MUST be one of the following values: ▪ 0x00000010: In ▪ 0x00000020: Out
0x0000000E	Random bars. The *corresponding effect direction* MUST be one of the following values: ▪ 0x0000000A: Horizontal ▪ 0x00000005: Vertical
0x0000000F	Spiral.

Value	Meaning
	The *corresponding effect direction* MUST be ignored.
0x00000010	Split. The *corresponding effect direction* MUST be one of the following values: ▪ 0x0000001A: Horizontal in ▪ 0x0000002A: Horizontal out ▪ 0x00000015: Vertical in ▪ 0x00000025: Vertical out
0x00000011	Stretch. The *corresponding effect direction* MUST be one of the following values: ▪ 0x00000002: Right ▪ 0x00000008: Left ▪ 0x00000001: Top ▪ 0x00000004: Bottom ▪ 0x0000000A: Across
0x00000012	Strips. The *corresponding effect direction* MUST be one of the following values: ▪ 0x00000009: Up left ▪ 0x00000003: Up right ▪ 0x00000006: Down right ▪ 0x0000000C: Down left
0x00000013	Swivel. The *corresponding effect direction* MUST be one of the following values: ▪ 0x0000000A: Horizontal ▪ 0x00000005: Vertical
0x00000014	Wedge.

Value	Meaning
	The *corresponding effect direction* MUST be ignored.
0x00000015	Wheel. The *corresponding effect direction* MUST be one of the following values: ▪ 0x00000001: Wheel(1) ▪ 0x00000002: Wheel(2) ▪ 0x00000003: Wheel(3) ▪ 0x00000004: Wheel(4) ▪ 0x00000008: Wheel(8)
0x00000016	Wipe. The *corresponding effect direction* MUST be one of the following values: ▪ 0x00000001: Up ▪ 0x00000002: Right ▪ 0x00000004: Down ▪ 0x00000008: Left
0x00000017	Zoom. The *corresponding effect direction* MUST be one of the following values: ▪ 0x00000010: In ▪ 0x00000020: Out ▪ 0x00000210: In center ▪ 0x00000024: Out bottom ▪ 0x00000120: Out slightly ▪ 0x00000110: In slightly
0x00000018	Random effects. The *corresponding effect direction* MUST be ignored.
0x00000019	Boomerang. The *corresponding effect direction* MUST be ignored.
0x0000001A	Bounce.

Value	Meaning
	The *corresponding effect direction* MUST be ignored.
0x0000001B	Color reveal. The *corresponding effect direction* MUST be ignored.
0x0000001C	Credits. The *corresponding effect direction* MUST be ignored.
0x0000001D	Ease in. The *corresponding effect direction* MUST be ignored.
0x0000001E	Float. The *corresponding effect direction* MUST be ignored.
0x0000001F	Grow and turn. The *corresponding effect direction* MUST be ignored.
0x00000020	Reserved.
0x00000021	Reserved.
0x00000022	Light speed. The *corresponding effect direction* MUST be ignored.
0x00000023	Pin wheel. The *corresponding effect direction* MUST be ignored.
0x00000024	Reserved.
0x00000025	Rise up. The *corresponding effect direction* MUST be ignored.
0x00000026	Swish. The *corresponding effect direction* MUST be ignored.
0x00000027	Thin line. The *corresponding effect direction* MUST be ignored.
0x00000028	Unfold. The *corresponding effect direction* MUST be ignored.
0x00000029	Whip. The *corresponding effect direction* MUST be ignored.

Value	Meaning
0x0000002A	Ascend. The *corresponding effect direction* MUST be ignored.
0x0000002B	Center revolve. The *corresponding effect direction* MUST be ignored.
0x0000002C	Reserved.
0x0000002D	Faded swivel. The *corresponding effect direction* MUST be ignored.
0x0000002E	Reserved.
0x0000002F	Descend. The *corresponding effect direction* MUST be ignored.
0x00000030	Sling. The *corresponding effect direction* MUST be ignored.
0x00000031	Spinner. The *corresponding effect direction* MUST be ignored.
0x00000032	Compress. The *corresponding effect direction* MUST be ignored.
0x00000033	Zip. The *corresponding effect direction* MUST be ignored.
0x00000034	Arc up. The *corresponding effect direction* MUST be ignored.
0x00000035	Faded zoom. The *corresponding effect direction* MUST be ignored.
0x00000036	Glide. The *corresponding effect direction* MUST be ignored.
0x00000037	Expand. The *corresponding effect direction* MUST be ignored.
0x00000038	Flip.

Value	Meaning
	The *corresponding effect direction* MUST be ignored.
0x00000039	Reserved.
0x0000003A	Fold. The *corresponding effect direction* MUST be ignored.

When the *corresponding effect type* is an emphasis effect, this field MUST be a value from the following table:

Value	Meaning
0x00000000	Custom. The *corresponding effect direction* MUST be ignored.
0x00000001	Change fill color. The *corresponding effect direction* MUST be one of the following values: ▪ 0x00000001: Instant ▪ 0x00000002: Gradual ▪ 0x00000006: Gradual and cycle clockwise ▪ 0x0000000A: Gradual and cycle counterclockwise
0x00000002	Change font. The *corresponding effect direction* MUST be ignored.
0x00000003	Change font color. The *corresponding effect direction* MUST be one of the following values: ▪ 0x00000001: Instant ▪ 0x00000001: Gradual ▪ 0x00000006: Gradual and cycle clockwise ▪ 0x0000000A: Gradual and cycle counterclockwise
0x00000004	Change font size. The *corresponding effect direction* MUST be one of the following values: ▪ 0x00000001: Instant ▪ 0x00000002: Gradual

Value	Meaning
0x00000005	Change font style. The *corresponding effect direction* MUST be a value of any combination of the following values: ▪ 0x00000001: Font bold ▪ 0x00000002: Font italic ▪ 0x00000004: Font underline
0x00000006	Grow and shrink. The *corresponding effect direction* MUST be ignored.
0x00000007	Change line color. The *corresponding effect direction* MUST be one of the following values: ▪ 0x00000001: Instant ▪ 0x00000002: Gradual ▪ 0x00000006: Gradual and cycle clockwise ▪ 0x0000000A: Gradual and cycle counterclockwise
0x00000008	Spin. The *corresponding effect direction* MUST be ignored.
0x00000009	Transparency. The *corresponding effect direction* MUST be ignored.
0x0000000A	Bold flash. The *corresponding effect direction* MUST be ignored.
0x0000000B	Reserved.
0x0000000C	Reserved.
0x0000000D	Reserved.
0x0000000E	Blast. The *corresponding effect direction* MUST be ignored.
0x0000000F	Bold reveal. The *corresponding effect direction* MUST be ignored.
0x00000010	Brush on color.

Value	Meaning
	The *corresponding effect direction* MUST be ignored.
0x00000011	Reserved.
0x00000012	Brush on underline. The *corresponding effect direction* MUST be ignored.
0x00000013	Color blend. The *corresponding effect direction* MUST be ignored.
0x00000014	Color wave. The *corresponding effect direction* MUST be ignored.
0x00000015	Complementary color. The *corresponding effect direction* MUST be ignored.
0x00000016	Complementary color 2. The *corresponding effect direction* MUST be ignored.
0x00000017	Contrasting color. The *corresponding effect direction* MUST be ignored.
0x00000018	Darken. The *corresponding effect direction* MUST be ignored.
0x00000019	Desaturate. The *corresponding effect direction* MUST be ignored.
0x0000001A	Flash bulb. The *corresponding effect direction* MUST be ignored.
0x0000001B	Flicker. The *corresponding effect direction* MUST be ignored.
0x0000001C	Grow with color. The *corresponding effect direction* MUST be ignored.
0x0000001D	Reserved.
0x0000001E	Lighten. The *corresponding effect direction* MUST be ignored.
0x0000001F	Style emphasis.

Value	Meaning
	The *corresponding effect direction* MUST be ignored.
0x00000020	Teeter. The *corresponding effect direction* MUST be ignored.
0x00000021	Vertical grow. The *corresponding effect direction* MUST be ignored.
0x00000022	Wave. The *corresponding effect direction* MUST be ignored.
0x00000023	Blink. The *corresponding effect direction* MUST be ignored.
0x00000024	Shimmer. The *corresponding effect direction* MUST be ignored.

When the *corresponding effect type* is a motion path effect, the *corresponding effect direction* MUST be ignored and this field MUST be a value from the following table:

Value	Meaning
0x00000000	Custom.
0x00000001	Circle.
0x00000002	Right triangle.
0x00000003	Diamond.
0x00000004	Hexagon.
0x00000005	5-point star.
0x00000006	Crescent moon.
0x00000007	Square.
0x00000008	Trapezoid.
0x00000009	Heart.
0x0000000A	Octagon.
0x0000000B	6-point star.
0x0000000C	Football.
0x0000000D	Equal triangle.
0x0000000E	Parallelogram.
0x0000000F	Pentagon.

Value	Meaning
0x00000010	4-point star.
0x00000011	8-point star.
0x00000012	Teardrop.
0x00000013	Pointy star.
0x00000014	Curved square.
0x00000015	Curved X.
0x00000016	Vertical figure 8.
0x00000017	Curvy star.
0x00000018	Loop de loop.
0x00000019	Buzz saw.
0x0000001A	Horizontal figure 8.
0x0000001B	Peanut.
0x0000001C	Figure 8 four.
0x0000001D	Neutron.
0x0000001E	Swoosh.
0x0000001F	Bean.
0x00000020	Plus.
0x00000021	Inverted triangle.
0x00000022	Inverted square.
0x00000023	Left.
0x00000024	Turn right.
0x00000025	Arc down.
0x00000026	Zigzag.
0x00000027	S curve 2.
0x00000028	Sine wave.
0x00000029	Bounce left.
0x0000002A	Down.
0x0000002B	Turn up.
0x0000002C	Arc up.
0x0000002D	Heartbeat.
0x0000002E	Spiral right.
0x0000002F	Wave.

Value	Meaning
0x00000030	Curvy left.
0x00000031	Diagonal down right.
0x00000032	Turn down.
0x00000033	Arc left.
0x00000034	Funnel.
0x00000035	Spring.
0x00000036	Bounce right.
0x00000037	Spiral left.
0x00000038	Diagonal up right.
0x00000039	Turn up right.
0x0000003A	Arc right.
0x0000003B	S curve 1.
0x0000003C	Decaying wave.
0x0000003D	Curvy right.
0x0000003E	Stairs down.
0x0000003F	Right.
0x00000040	Up.

When the *corresponding effect type* is a media effect, the *corresponding effect direction* MUST be ignored and this field MUST be a value from the following table:

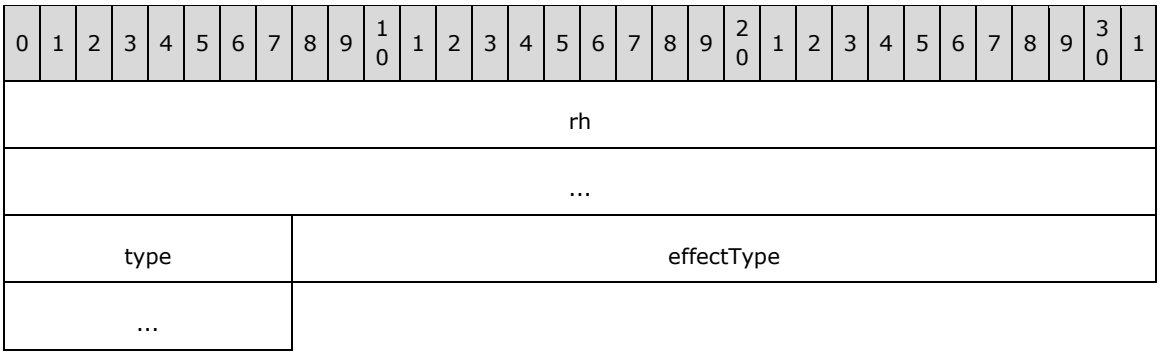
Value	Meaning
0x00000000	Custom.
0x00000001	Play.
0x00000002	Pause.
0x00000003	Stop.

2.8.24 TimeEffectType

Referenced by: TimeVariant4TimeNode

An atom record that specifies the type of animation effect.

Let the *corresponding time node* be as specified in the **TimePropertyList4TimeNodeContainer** record (section 2.8.18) that contains this **TimeEffectType** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be an RT_TimeVariant.
rh.recLen	MUST be 0x00000005.

type (1 byte): A **TimeVariantTypeEnum** enumeration that specifies the data type of this record. It MUST be TL_TVT_Int.

effectType (4 bytes): A **signed integer** that specifies the type of animation effect of the *corresponding time node*. It MUST be a value from the following table.

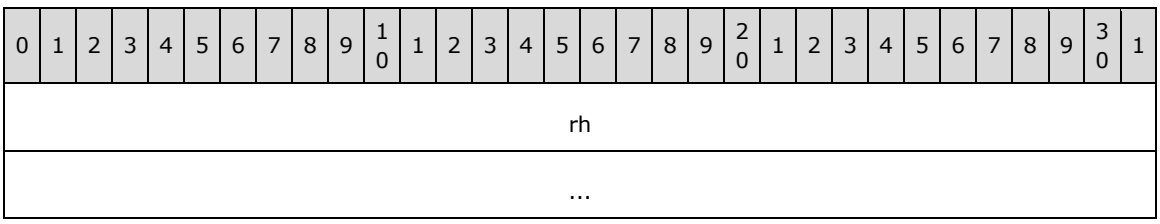
Value	Meaning
0x00000001	Entrance.
0x00000002	Exit.
0x00000003	Emphasis.
0x00000004	Motion path.
0x00000005	Action verb.
0x00000006	Media command.

2.8.25 TimeNodeTimeFilter

Referenced by: TimeVariant4TimeNode

An atom record that specifies a time filter that transforms a given time to the local time of a time node.

Let the *corresponding time node* be as specified in the **TimePropertyList4TimeNodeContainer** record (section 2.8.18) that contains this **TimeNodeTimeFilter** record.



type	timeFilter (variable)
...	

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be an RT_TimeVariant.
rh.recLen	MUST be an odd number.

type (1 byte): A **TimeVariantTypeEnum** enumeration that specifies the data type of this record. It MUST be TL_TVT_String.

timeFilter (variable): A UnicodeString that specifies the time filter that transforms a given time value into the local time of the *corresponding time node*. It MUST be a valid TIMESEQUENCE as specified in the following ABNF (specified in [RFC5234]) grammar:

```
TIMESEQUENCE = TIME "," TRANSTIME *(";" TIMESEQUENCE)
TIME = "0" "." 1*DIGIT / "1" "." "0"
TRANSTIME = "0" "." 1*DIGIT / "1" "." "0"
```

Each TIME is the normalized local time for the time node whose value ranges from 0.0 to 1.0, and TRANSTIME is the transformed local time for the time node. The length, in bytes, of the field is specified by the following formula:

```
rh.recLen - 1
```

2.8.26 TimeEventFilter

Referenced by: TimeVariant4TimeNode

An atom record that specifies an event filter for a time node.

Let the *corresponding time node* be as specified in the **TimePropertyList4TimeNodeContainer** record (section 2.8.18) that contains this **TimeEventFilter** record.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
type											timeEventFilter (variable)																				
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be an RT_TimeVariant.
rh.recLen	MUST be an odd number.

type (1 byte): A **TimeVariantTypeEnum** enumeration that specifies the data type of this record. It MUST be TL_TVT_String.

timeEventFilter (variable): A UnicodeString that specifies the event filter for the *corresponding time node*. It MUST be "cancelBubble" and the **TimePropertyList4TimeNodeContainer** record that contains this **TimeEventFilter** record MUST contain a TimeEffectNodeType record with **effectNodeType** equal to 0x000000005. The length, in bytes, of the field is specified by the following formula:

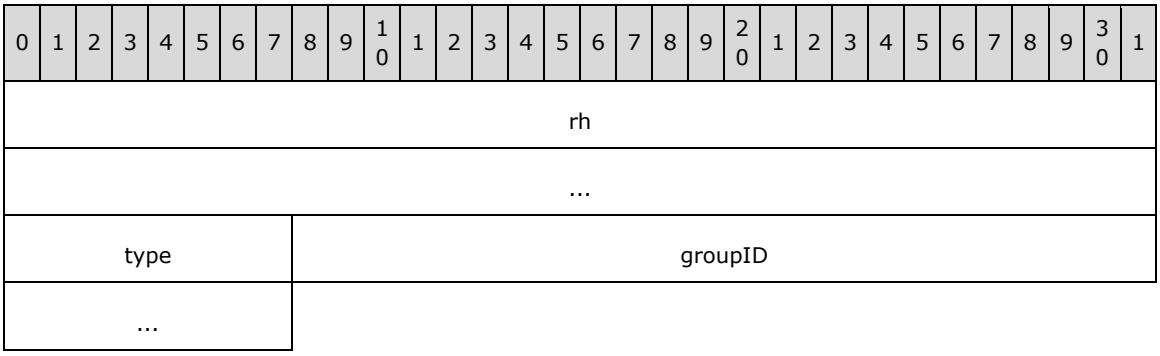
```
rh.recLen - 1
```

2.8.27 TimeGroupID

Referenced by: TimeVariant4TimeNode

An atom record that specifies a reference to a build identifier of an animation effect.

Let the *corresponding time node* be as specified in the **TimePropertyList4TimeNodeContainer** record (section 2.8.18) that contains this **TimeGroupID** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be an RT_TimeVariant.
rh.recLen	MUST be 0x00000005.

type (1 byte): A **TimeVariantTypeEnum** enumeration that specifies the data type of this record. It MUST be TL_TVT_Int.

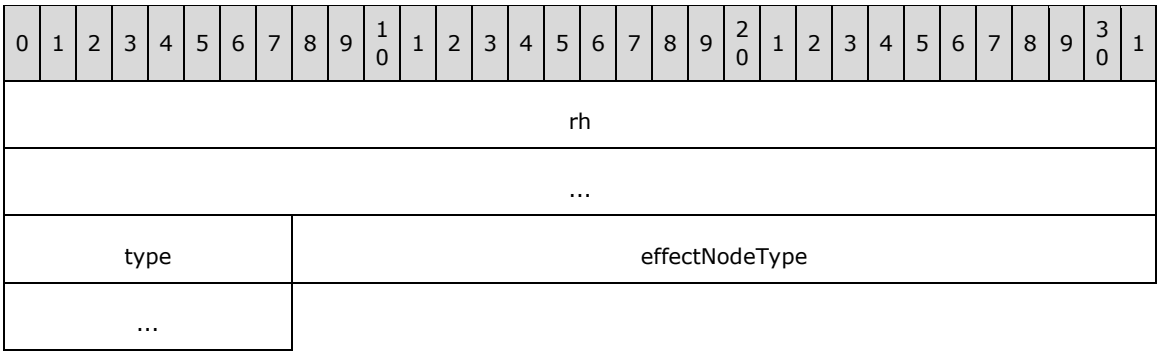
groupID (4 bytes): A **signed integer** that specifies a reference to a build identifier of the animation as specified in the BuildAtom record. There is a **shapeIdRef** field specified in either the VisualShapeChartElementAtom record or the VisualShapeGeneralAtom record; it is used with this field to form a unique pair for the animation effect that will be applied to the specified object.

2.8.28 TimeEffectNodeType

Referenced by: TimeVariant4TimeNode

An atom record that specifies the role of a time node in the timing structure.

Let the *corresponding time node* be as specified in the **TimePropertyList4TimeNodeContainer** record (section 2.8.18) that contains this **TimeEffectNodeType** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be an RT_TimeVariant.
rh.recLen	MUST be 0x00000005.

type (1 byte): A **TimeVariantTypeEnum** enumeration that specifies the data type of this record. It MUST be TL_TVT_Int.

effectNodeType (4 bytes): A **signed integer** that specifies the role of the *corresponding time node*. It MUST be a value from the following table.

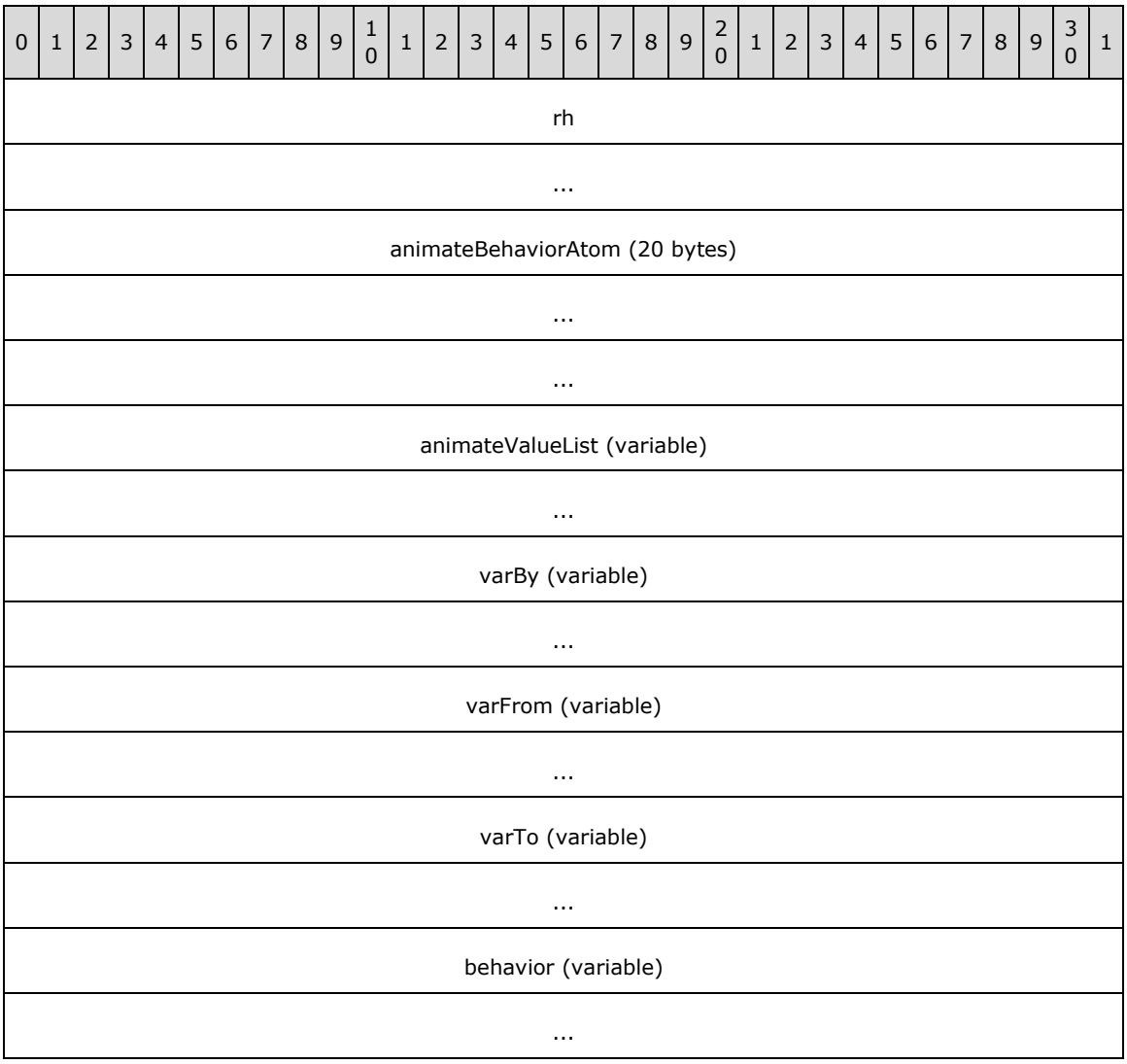
Value	Meaning
0x00000001	Click effect node.
0x00000002	With previous node.
0x00000003	After previous node.
0x00000004	Main sequence node.
0x00000005	Interactive sequence node.
0x00000006	Click parallel node.
0x00000007	With group node.
0x00000008	After group node.
0x00000009	Timing root node.

2.8.29 TimeAnimateBehaviorContainer

Referenced by: ExtTimeNodeContainer

A container record that specifies a generic animation behavior. This animation behavior is applied to the object specified by the **behavior.clientVisualElement** field and used to animate one

property specified by the **behavior.stringList** field. The property MUST be one from the list that is specified in the **TimeStringListContainer** record (section 2.8.36).



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeAnimateBehaviorContainer.

animateBehaviorAtom (20 bytes): A TimeAnimateBehaviorAtom record that specifies the value type of the animated property as specified in the **behavior.stringList** field, how to interpolate the value, and which attributes of this field and this **TimeAnimateBehaviorContainer** record are valid.

animateValueList (variable): An optional **TimeAnimationValueListContainer** record (section 2.8.31) that specifies the list of key points that consists of a time and the value at that time. The **animateBehaviorAtom.calcMode** field specifies how to calculate interpolated values between

these key points. It MUST be ignored if **animateBehaviorAtom.fAnimationValuesPropertyUsed** is **FALSE**.

varBy (variable): An optional TimeVariantString record that specifies the offset value of the animated property. The sub-field **varBy.rh.recInstance** MUST be 0x001. It MUST be in a format dictated by the value of **animateBehaviorAtom.valueType** as specified in the following table.

Value	Meaning
TL_TABVT_String	An arbitrary Unicode string.
TL_TABVT_Number	A preset string value as specified in the **varTo** field of the **TimeSetBehaviorContainer** record (section 2.8.69); or a string that specifies a real number, whose format is specified by the **varTo** field of the **TimeSetBehaviorContainer** record.
TL_TABVT_Color	A string that specifies a color value, whose format is specified by the **varTo** field of the **TimeSetBehaviorContainer** record.

MUST be ignored if **varTo** exists. It MUST be ignored if **animateValueList** exists. It MUST be ignored if **animateBehaviorAtom.fByPropertyUsed** is **FALSE**.

varFrom (variable): An optional TimeVariantString record that specifies the starting value of the animated property. The sub-field **varFrom.rh.recInstance** MUST be 0x002. It MUST be in a format dictated by the value of **animateBehaviorAtom.valueType** as specified in the following table.

Value	Meaning
TL_TABVT_String	An arbitrary Unicode string.
TL_TABVT_Number	A preset string value as specified in the **varTo** field of the **TimeSetBehaviorContainer** record; or a string that specifies a real number, whose format is specified by the **varTo** field of the **TimeSetBehaviorContainer** record.
TL_TABVT_Color	A string that specifies a color value, whose format is specified by the **varTo** field of the **TimeSetBehaviorContainer** record.

If **varFrom** exists, **varTo** or **varBy** MUST also exist. It MUST be ignored if **animateValueList** exists. It MUST be ignored if **animateBehaviorAtom.fFromPropertyUsed** is **FALSE**.

varTo (variable): An optional TimeVariantString record that specifies the end value of the animated property. The sub-field **varTo.rh.recInstance** MUST be 0x003. It MUST be in a format dictated by the value of **animateBehaviorAtom.valueType** as specified in the following table.

Value	Meaning
TL_TABVT_String	An arbitrary Unicode string.
TL_TABVT_Number	A preset string value as specified in the **varTo** field of the **TimeSetBehaviorContainer** record; or a string that specifies a real number, whose format is specified by the **varTo** field of the **TimeSetBehaviorContainer** record.
TL_TABVT_Color	A string that specifies a color value, whose format is specified by the **varTo** field of the **TimeSetBehaviorContainer** record.

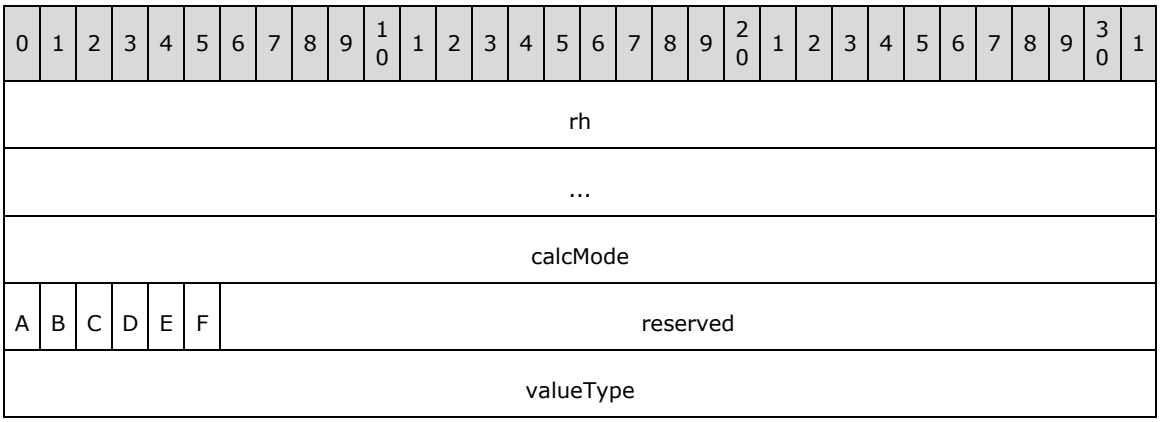
MUST be ignored if **animateValueList** exists. It MUST be ignored if **animateBehaviorAtom.fToPropertyUsed** is **FALSE**.

behavior (variable): A **TimeBehaviorContainer** record (section 2.8.34) that specifies the common animation behavior information.

2.8.30 TimeAnimateBehaviorAtom

Referenced by: TimeAnimateBehaviorContainer

An atom record that specifies the information of a generic animation that can animate any property.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeAnimateBehavior.
rh.recLen	MUST be 0x0000000C.

calcMode (4 bytes): An **unsigned integer** that specifies how the property value is calculated. It MUST be ignored if **fCalcModePropertyUsed** is **FALSE** and a value of 0x00000001 MUST be used instead. It MUST be a value from the following table:

Value	Meaning
0x00000000	Discrete mode, which specifies that the value will jump from one to another without any interpolation.
0x00000001	Linear mode, which specifies that the values are linearly interpolated.
0x00000002	Formula mode, which specifies that a formula specified by the **animateValueList** field of the **TimeAnimateBehaviorContainer** record (section 2.8.29) that contains this **TimeAnimateBehaviorAtom** record is used in the interpolation.

A - fByPropertyUsed (1 bit): A bit that specifies whether the **varBy** field of the **TimeAnimateBehaviorContainer** record (section 2.8.29) that contains this **TimeAnimateBehaviorAtom** record is valid.

B - fFromPropertyUsed (1 bit): A bit that specifies whether the **varFrom** field of the **TimeAnimateBehaviorContainer** record (section 2.8.29) that contains this **TimeAnimateBehaviorAtom** record is valid.

C - fToPropertyUsed (1 bit): A bit that specifies whether the **varTo** field of the **TimeAnimateBehaviorContainer** record (section 2.8.29) that contains this **TimeAnimateBehaviorAtom** record is valid.

D - fCalcModePropertyUsed (1 bit): A bit that specifies whether **calcMode** was explicitly set by a user interface action.

E - fAnimationValuesPropertyUsed (1 bit): A bit that specifies whether the **animateValueList** field of the **TimeAnimateBehaviorContainer** record (section 2.8.29) that contains this **TimeAnimateBehaviorAtom** record is valid.

F - fValueTypePropertyUsed (1 bit): A bit that specifies whether **valueType** was explicitly set by a user interface action.

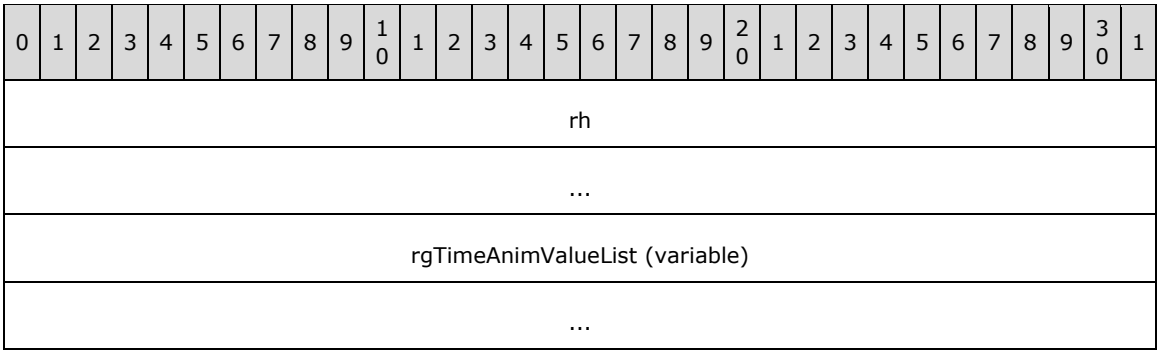
reserved (26 bits): MUST be zero, and MUST be ignored.

valueType (4 bytes): A **TimeAnimateBehaviorValueTypeEnum** enumeration that specifies the value type of the property to be animated. It MUST be ignored if **fValueTypePropertyUsed** is **FALSE** and a value of TL_TABVT_Number MUST be used instead.

2.8.31 TimeAnimationValueListContainer

Referenced by: TimeAnimateBehaviorContainer

A container record that specifies the list of key points that are used during a property animation to set a property to a value at a point within the timeline as specified in the **TimeAnimateBehaviorContainer** record (section 2.8.29).



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeAnimationValueList.

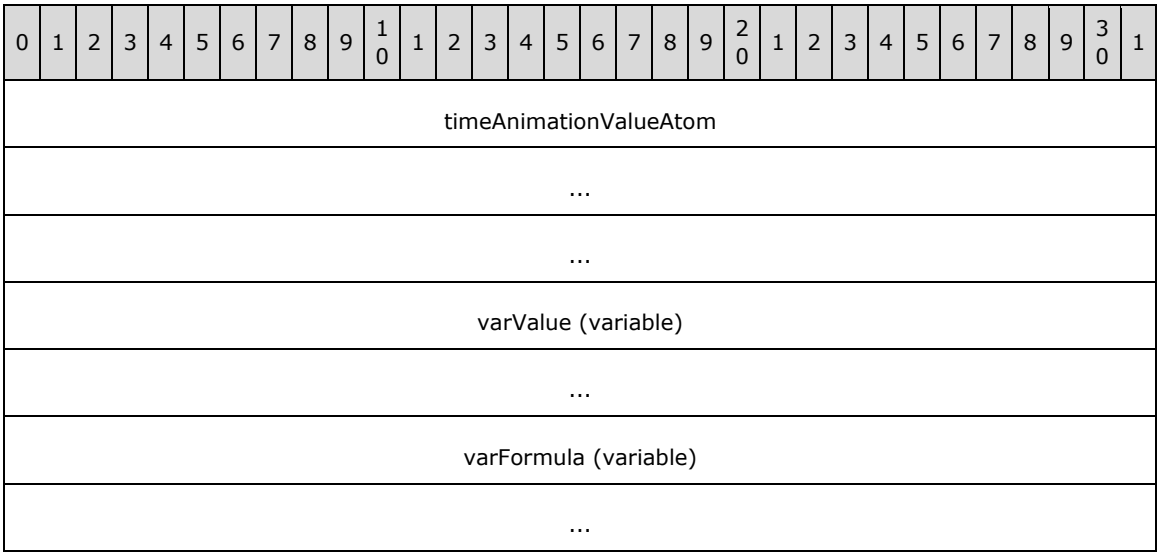
rgTimeAnimValueList (variable): An array of TimeAnimationValueListEntry structures that specifies the key points that are used during the animation. The length, in bytes, of the array is specified by **rh.recLen**.

If the value of the **timeAnimationValueAtom.time** field in any TimeAnimationValueListEntry record in this array is -1000, the time for the key points is evenly partitioned between 0 and 1.

2.8.32 TimeAnimationValueListEntry

Referenced by: TimeAnimationValueListContainer

A structure that specifies a key point in a property animation.



timeAnimationValueAtom (12 bytes): A TimeAnimationValueAtom record that specifies the time, as a percentage of the animation time, at which the property takes on the value specified by the **varValue** and **varFormula** fields.

varValue (variable): An optional TimeVariant record that specifies a value that corresponds to the time as specified in the **timeAnimationValueAtom.time** field. The sub-field **varValue.rh.recInstance** MUST be 0x000. If the **varFormula** field exists, the property takes on the value specified by the formula; otherwise, the property takes on this value.

varFormula (variable): An optional TimeVariantString record that specifies a formula to be used to specify a complex animation for an object. The sub-field **varFormula.rh.recInstance** MUST be 0x001.

The formula manipulates a property value of the object, over a specified period of time. Each formula has zero or more inputs specified by the ($) symbol, zero or more variables specified by the (#) symbol, and a target variable. In each instance, the special symbols are pre-pended to the variable name. The target variable is specified by the **behavior.stringList.rgChildRec** field of the **TimeAnimateBehaviorContainer** record (section 2.8.29) that also contains the **TimeAnimationValueListContainer** record (section 2.8.31) that contains this **TimeAnimationValueListEntry** record. The formula may contain one or more of any of the following constants, operators or functions. In addition, the formula may also contain **floating-point numbers** and parentheses.

The **varFormula.stringValue** MUST be a valid FORMULA as specified in the following ABNF (specified in [RFC5234]) grammar:

```
FORMULA = TERM *(( "+" / "-" ) TERM)
TERM = POWER *(( "*" / "/" / "%" ) POWER)
POWER = UNARY *("^" UNARY)
UNARY = ["+" / "-"] FACTOR
FACTOR = VARIABLE / CONSTANT / FUNCTION / PARENS
PARENS = "(" FORMULA ")"CHAR = "." / " " / ALPHA / DIGIT
NUMBER   = 1*DIGIT
EXPONENT = ( "e" / "E" ) ["-"] NUMBER
VALUE = NUMBER ["." NUMBER] [EXPONENT]
VARIABLE = "$" / ATTRIBUTE
ATTRIBUTE = ["#"] ATTRNAME
ATTRNAME = OFFICEART_CLIENT_ATTRNAME / OFFICEART_FOPT_ATTRNAME
OFFICEART_CLIENT_ATTRNAME = "ppt_x" / "ppt_y" / "ppt_w" / "ppt_h" / "ScaleX" / "ScaleY"
OFFICEART_FOPT_ATTRNAME = "stype.rotation" / "style.opacity" / "style.visibility" / "ppt_r" /
"r" / "style.fontSize" / "style.fontWeight" / "style.fontStyle" / "style.fontFamily" /
"style.textEffectEmboss" / "style.textShadow" / "style.textTransform" /
"style.textDecorationUnderline" / "style.textEffectOutline" /
"style.textDecorationLineThrough" / "style.sRotation" / "imageData.cropTop" /
"imageData.cropBottom" / "imageData.cropLeft" / "imageData.cropRight" / "imageData.gain" /
"imageData.blackleve" / "imageData.gamma" / "imageData.grayscale" / "fill.on" / "fill.type" /
"fill.opacity" / "fill.method" / "fill.opacity2" / "fill.angle" / "fill.focus" /
"fill.focusposition.x" / "fill.focusposition.y" / "fill.focussize.x" / "fill.focussize.y" /
"stroke.on" / "stroke.weight" / "stroke.opacity" / "stroke.linestyle" / "stroke.dashstyle" /
"stroke.filltype" / "stroke.imagesize.x" / "stroke.imagesize.y" / "stroke.startArrow" /
"stroke.endArrow" / "stroke.startArrowWidth" / "stroke.startArrowLength" /
"stroke.endArrowWidth" / "stroke.endArrowLength" / "shadow.on" / "shadow.type" /
"shadow.opacity" / "shadow.offset.x" / "shadow.offset.y" / "shadow.offset2.x" /
"shadow.offset2.y" / "shadow.origin.x" / "shadow.origin.y" / "shadow.matrix.xtox" /
"shadow.matrix.ytox" / "shadow.matrix.xtox" / "shadow.matrix.ytoy" /
"shadow.matrix.perspectiveX" / "shadow.matrix.perspectiveY" / "skew.on" / "skew.offset.x" /
"skew.offset.y" / "skew.origin.x" / "skew.origin.y" / "skew.matrix.xtox" / "skew.matrix.ytox"
/ "skew.matrix.xtox" / "skew.matrix.ytoy" / "skew.matrix.perspectiveX" /
"skew.matrix.perspectiveY" / "extrusion.on" / "extrusion.type" / "extrusion.render" /
"extrusion.viewpointorigin.x" / "extrusion.viewpointorigin.y" / "extrusion.viewpoint.x" /
"extrusion.viewpoint.y" / "extrusion.viewpoint.z" / "extrusion.plane" / "extrusion.skewangle"
/ "extrusion.skewamt" / "extrusion.backdepth" / "extrusion.foredepth" /
"extrusion.orientation.x" / "extrusion.orientation.y" / "extrusion.orientation.z" /
"extrusion.orientationangle" / "extrusion.rotationangle.x" / "extrusion.rotationangle.y" /
"extrusion.lockrotationcenter" / "extrusion.autorotationcenter" /
"extrusion.rotationcenter.x" / "extrusion.rotationcenter.y" / "extrusion.rotationcenter.z" /
"extrusion.colormode"
CONSTANT = VALUE / "pi" / "e"
IDENT = "abs" / "acos" / "asin" / "atan" / "ceil" / "cos" / "cosh" / "deg" / "exp" / "floor"
/ "ln" / "max" / "min" / "rad" / "rand" / "sin" / "sinh" / "sqrt" / "tan" / "tanh"
FUNCTION = IDENT "(" FORMULA ["," FORMULA] ")"
```

Components of the preceding formula are further specified as follows.

Operator precedence

Mathematical operations have the following order of precedence, listed from lowest to highest. Operators listed on the same line have equal precedence.

1. "+" (Addition), "-" (Subtraction)
2. "*" (Multiplication), "/" (Division), "%" (Modulo)
3. "^" (Exponentiation)
4. "-" (Unary minus), "+" (Unary plus)
5. Variables, Constants (including numbers) and Functions

Variables

The symbol '$' represents the value of **varValue**.

Attributes

ATTRNAME MUST be one from the following two lists:

- OFFICEART_CLIENT_ATTRNAME specifies the list of attributes that are translated from the OfficeArtClientAnchor record.
- OFFICEART_FOPT_ATTRNAME specifies the list of attributes that are translated from the OfficeArtFOPT record ([MS-ODRAW] section 2.2.9).

Constants

Name	Description
pi	Mathematical constant pi
e	Mathematical constant e

Operators

Name	Description	Usage
+	Addition	"x+y", adds x to the value y
-	Subtraction	"x-y", subtracts y from the value x
*	Multiplication	"x*y", multiplies x by the value y
/	Division	"x/y", divides x by the value y
%	Modulus	"x%y", the remainder of x/y
^	Power	"x^y", x raised to the power y

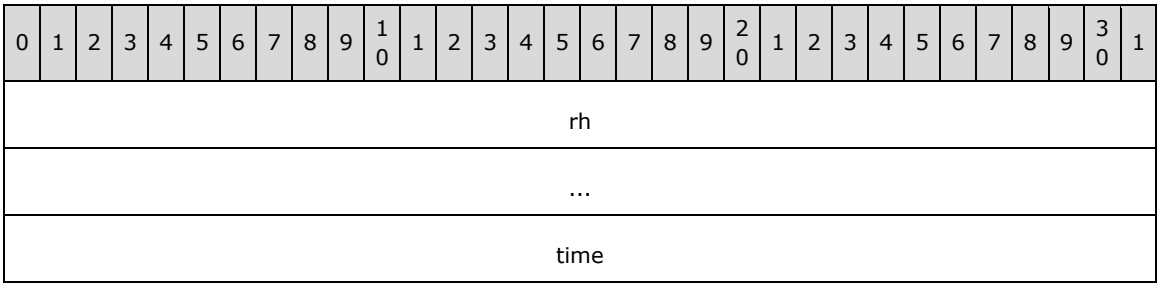
Functions

Name	Description	Usage
abs	Absolute value	"abs(x)", absolute value of x
acos	Arc Cosine	"acos(x)", arc cosine of the value x
asin	Arc Sine	"asin(x)", arc sine of the value x
atan	Arc Tangent	"atan(x)", arc tangent of the value x
ceil	Ceil value	"ceil(x)", value of x rounded up
cos	Cosine	"cos(x)", cosine of the value of x
cosh	Hyperbolic Cosine	"cosh(x)", hyperbolic cosine of the value x
deg	Radiant to Degree convert	"deg(x)", the degree value of radiant value x
exp	Exponent	"exp(x)", value of constant e raised to the power of x
floor	Floor value	"floor(x)", value of x rounded down
ln	Natural logarithm	"ln(x)", natural logarithm of x
max	Maximum of two values	"max(x,y)", returns x if (x > y) or returns y if (y > x)
min	Minimum of two values	"min(x,y)", returns x if (x < y) or returns y if (y < x)
rad	Degree to Radiant convert	"rad(x)", the radiant value of degree value x
rand	Random value	"rand(x)", returns a random floating point value between 0 and x
sin	Sine	"sin(x)", sine of the value x
sinh	Hyperbolic Sine	"sinh(x)", hyperbolic sine of the value x
sqrt	Square root	"sqrt(x)", square root of the value x
tan	Tangent	"tan(x)", tangent of the value x
tanh	Hyperbolic Tangent	"tanh(x)", hyperbolic tangent of the value x

2.8.33 TimeAnimationValueAtom

Referenced by: TimeAnimationValueListEntry

An atom record that specifies a value of time that is used in the TimeAnimationValueListEntry structure to determine the overall timeline for the corresponding animation.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

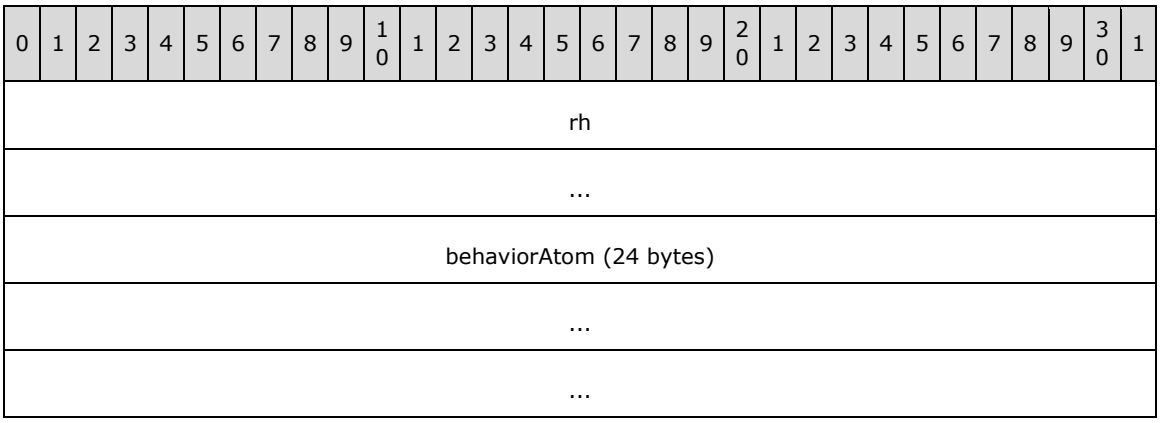
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeAnimationValue.
rh.recLen	MUST be 0x00000004.

time (4 bytes): A **signed integer** that specifies a percentage value of time in 1000ths that is utilized in the TimeAnimationValueListEntry record. For example, 1000 means 100% of time of an animation. This field MUST be either equal to -1000, or greater than or equal to 0 and less than or equal to 1000.

2.8.34 TimeBehaviorContainer

Referenced by: TimeAnimateBehaviorContainer, TimeColorBehaviorContainer, TimeCommandBehaviorContainer, TimeEffectBehaviorContainer, TimeMotionBehaviorContainer, TimeRotationBehaviorContainer, TimeScaleBehaviorContainer, TimeSetBehaviorContainer

A container record that specifies the common information of an animation behavior.



stringList (variable)
...
propertyList (variable)
...
clientVisualElement (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_TimeBehaviorContainer.

behaviorAtom (24 bytes): A TimeBehaviorAtom record that specifies the type of transform used in the animation, how to compose animated values, and which attributes of this field and this **TimeBehaviorContainer** record are valid.

stringList (variable): An optional **TimeStringListContainer** record (section 2.8.36) that specifies the list of the names of properties to be animated. It MUST be ignored if **behaviorAtom.fAttributeNamesPropertyUsed** is **FALSE**.

propertyList (variable): An optional TimePropertyList2.8.37TimeBehavior record that specifies a list of animation attributes that are used in the animation behavior.

clientVisualElement (variable): A **ClientVisualElementContainer** record (section 2.8.44) that specifies the target object that is animated.

2.8.35 TimeBehaviorAtom

Referenced by: TimeBehaviorContainer

An atom record that specifies the common information of an animation behavior.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
A	B	C	D	reserved3																											
behaviorAdditive																															

behaviorAccumulate
behaviorTransform

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeBehavior.
rh.recLen	MUST be 0x00000010.

A - fAdditivePropertyUsed (1 bit): A bit that specifies whether the **behaviorAdditive** field was explicitly set by a user interface action.

B - reserved1 (1 bit): MUST be zero, and MUST be ignored.

C - fAttributeNamesPropertyUsed (1 bit): A bit that specifies whether the **stringList** field of the **TimeBehaviorContainer** record (section 2.8.34) that contains this TimeBehaviorAtom record is valid.

D - reserved2 (1 bit): MUST be zero, and MUST be ignored.

reserved3 (28 bits): MUST be zero, and MUST be ignored.

behaviorAdditive (4 bytes): An unsigned integer that specifies how to compose the animated value with the original value of the property that is animated. It MUST be ignored if **fAdditivePropertyUsed** is **FALSE** and a value of 0x00000000 MUST be used instead. It MUST be a value from the following table.

Value	Meaning
0x00000000	Override the original value with the animated value.
0x00000001	Add the animated value to the original value.

behaviorAccumulate (4 bytes): An unsigned integer that specifies how to compose the animated values of the property in repeating animations. It MUST be 0x00000000 that specifies that no accumulation is used.

behaviorTransform (4 bytes): An unsigned integer that specifies the type of animation transform to use. It MUST be 0x00000000 that specifies that the animation animates properties of the target object.

2.8.36 TimeStringListContainer

Referenced by: TimeBehaviorContainer

A container record that specifies a list of names of properties that are animated by an animation behavior.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
rgChildRec (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be an RT_TimeVariantList.

rgChildRec (variable): An array of TimeVariantString record that specifies the list of names. The length, in bytes, of the array is specified by **rh.recLen**. The property names SHOULD be from the following list of possible names:

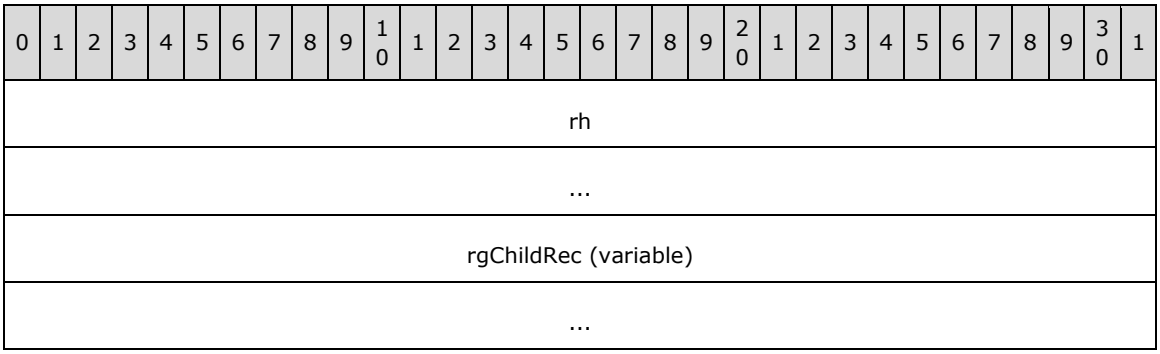
ppt_x, ppt_y, ppt_w, ppt_h, ppt_c, ppt_r, xshear, yshear, image, ScaleX, ScaleY, r, fillcolor, style.opacity, style.rotation, style.visibility, style.color, style.fontSize, style.fontWeight, style.fontStyle, style.fontFamily, style.textEffectEmboss, style.textShadow, style.textTransform, style.textDecorationUnderline, style.textEffectOutline, style.textDecorationLineThrough, style.sRotation, imageData.cropTop, imageData.cropBottom, imageData.cropLeft, imageData.cropRight, imageData.gain, imageData.blacklevel, imageData.gamma, imageData.grayscale, imageData.chromakey, fill.on, fill.type, fill.color, fill.opacity, fill.color2, fill.method, fill.opacity2, fill.angle, fill.focus, fill.focusposition.x, fill.focusposition.y, fill.focussize.x, fill.focussize.y, stroke.on, stroke.color, stroke.weight, stroke.opacity, stroke.linestyle, stroke.dashstyle, stroke.filltype, stroke.src, stroke.color2, stroke.imagesize.x, stroke.imagesize.y, stroke.startArrow, stroke.endArrow, stroke.startArrowWidth, stroke.startArrowLength, stroke.endArrowWidth, stroke.endArrowLength, shadow.on, shadow.type, shadow.color, shadow.color2, shadow.opacity, shadow.offset.x, shadow.offset.y, shadow.offset2.x, shadow.offset2.y, shadow.origin.x, shadow.origin.y, shadow.matrix.xtox, shadow.matrix.ytox, shadow.matrix.xtox, shadow.matrix.ytoy, shadow.matrix.perspectiveX, shadow.matrix.perspectiveY, skew.on, skew.offset.x, skew.offset.y, skew.origin.x, skew.origin.y, skew.matrix.xtox, skew.matrix.ytox, skew.matrix.xtox, skew.matrix.ytoy, skew.matrix.perspectiveX, skew.matrix.perspectiveY, extrusion.on, extrusion.type, extrusion.render, extrusion.viewpointorigin.x, extrusion.viewpointorigin.y, extrusion.viewpoint.x, extrusion.viewpoint.y, extrusion.viewpoint.z, extrusion.plane, extrusion.skewangle, extrusion.skewamt, extrusion.backdepth, extrusion.foredepth, extrusion.orientation.x, extrusion.orientation.y, extrusion.orientation.z, extrusion.orientationangle, extrusion.color, extrusion.rotationangle.x, extrusion.rotationangle.y, extrusion.lockrotationcenter, extrusion.autorotationcenter, extrusion.rotationcenter.x, extrusion.rotationcenter.y, extrusion.rotationcenter.z, and extrusion.colormode

2.8.37 TimePropertyList4TimeBehavior

Referenced by: TimeBehaviorContainer

A container record that specifies a list of animation attributes that is used in an animation behavior.

Let the *corresponding time node* be specified by the **ExtTimeNodeContainer** record (section 2.8.15) or the **SubEffectContainer** record (section 2.8.16) that contains this **TimePropertyList4TimeBehavior** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimePropertyList.

rgChildRec (variable): An array of TimeVariant2.8.38Behavior records that specifies the list of animation attributes. The length, in bytes, of the array is specified by **rh.recLen**. Each **TimePropertyID2.13.37TimeBehavior** enumeration value MUST NOT occur more than once as a value of the **rh.recInstance** field in the array.

If the TL_TBPID_MotionPathEditRelative value does not occur, a TimeVariantBool structure in which the **boolValue** field is 0x01 SHOULD be used.

If the TL_TBPID_PathEditRotationAngle value does not occur, a TimeVariantFloat structure in which the **floatValue** field is 0 SHOULD be used.

If the TL_TBPID_PathEditRotationX value does not occur, a TimeVariantFloat structure in which the **floatValue** field is 0 SHOULD be used.

If the TL_TBPID_PathEditRotationY value does not occur, a TimeVariantFloat structure in which the **floatValue** field is 0 SHOULD be used.

If the **colorBehaviorAtom.flags.fColorSpacePropertyUsed** field of the **TimeColorBehaviorContainer** record (section 2.8.52) is **FALSE**, any item with the TL_TBPID_ColorColorModel value MUST be ignored and a TimeColorModel record with a **colorModel** field equal to 0x00000000 MUST be used instead.

If the **colorBehaviorAtom.flags.fDirectionPropertyUsed** field of the **TimeColorBehaviorContainer** record is **FALSE**, any item with the TL_TBPID_ColorDirection value MUST be ignored and a TimeColorDirection record with a **colorDirection** field equal to 0x00000000 MUST be used instead.

2.8.38 TimeVariant4Behavior

Referenced by: TimePropertyList4TimeBehavior

A variable type record that specifies an attribute of an animation behavior and whose type and meaning are dictated by the value of the **rh.recInstance** field of any of these attributes, as specified in the following table.

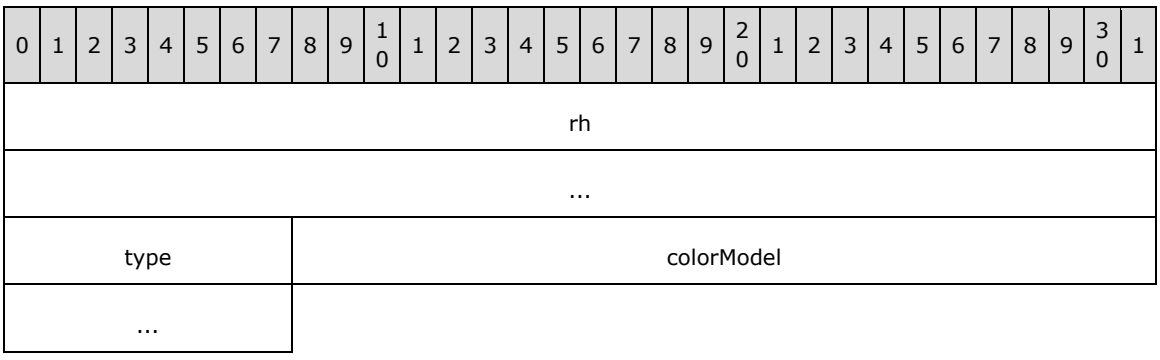
Value	Meaning
TL_TBPID_UnknownPropertyList	A TimeVariantString record that specifies unknown attributes.
TL_TBPID_RuntimeContext	A TimeRuntimeContext record that specifies which versions of the hosting applications can run this behavior.
TL_TBPID_MotionPathEditRelative	A TimeVariantBool record that specifies whether a motion path moves along with the object that it applies to during editing. This record is only used by the **TimeMotionBehaviorContainer** record (section 2.8.63).
TL_TBPID_ColorColorModel	A TimeColorModel record that specifies the color model. This record is only used by the **TimeColorBehaviorContainer** record (section 2.8.52).
TL_TBPID_ColorDirection	A TimeColorDirection record that specifies the interpolation direction in the **HSL color space**. This record is only used by the **TimeColorBehaviorContainer** record.
TL_TBPID_Override	A TimeOverride record that specifies how to override animated values.
TL_TBPID_PathEditRotationAngle	A TimeVariantFloat record that specifies the rotation angle of a motion path. This record is only used by the **TimeMotionBehaviorContainer** record.
TL_TBPID_PathEditRotationX	A TimeVariantFloat record that specifies the horizontal position of the rotation center of a motion path. This record is only used by the **TimeMotionBehaviorContainer** record.
TL_TBPID_PathEditRotationY	A TimeVariantFloat record that specifies the vertical position of the rotation center of a motion path. This record is only used by the **TimeMotionBehaviorContainer** record.
TL_TBPID_PointsTypes	A TimePointsTypes record that specifies the type of points in a motion path. This record is only used by the **TimeMotionBehaviorContainer** record.

2.8.39 TimeColorModel

Referenced by: TimeVariant4Behavior

An atom record that specifies the color model used by a color animation.

Let the *corresponding time node* be as specified in the TimePropertyList2.8.37TimeBehavior record that contains this **TimeColorModel** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be RT_TimeVariant.
rh.recLen	MUST be 0x00000005.

type (1 byte): A **TimeVariantTypeEnum** enumeration that specifies the data type of this record. It MUST be TL_TVT_Int.

colorModel (4 bytes): A signed integer that specifies the color model used by the color animation of the *corresponding time node*. It MUST be a value from the following table.

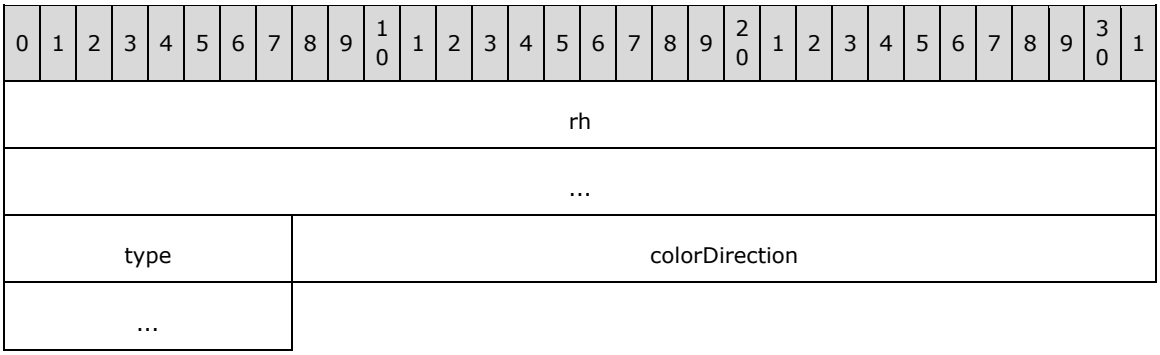
Value	Meaning
0x00000000	Use red, green, and blue color components in **red-green-blue (RGB)** color space.
0x00000001	Use hue, saturation, and luminance color components in HSL color space.
0x00000002	Use index to color scheme.

2.8.40 TimeColorDirection

Referenced by: TimeVariant4Behavior

An atom record that specifies the interpolation direction of a color animation.

Let the *corresponding time node* be as specified in the TimePropertyList2.8.37TimeBehavior record that contains this **TimeColorDirection** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be an RT_TimeVariant.
rh.recLen	MUST be 0x00000005.

type (1 byte): A **TimeVariantTypeEnum** enumeration that specifies the data type of this record. It MUST be TL_TVT_Int.

colorDirection (4 bytes): A **signed integer** that specifies the interpolation direction of the color animation of the *corresponding time node*. It MUST be a value from the following table.

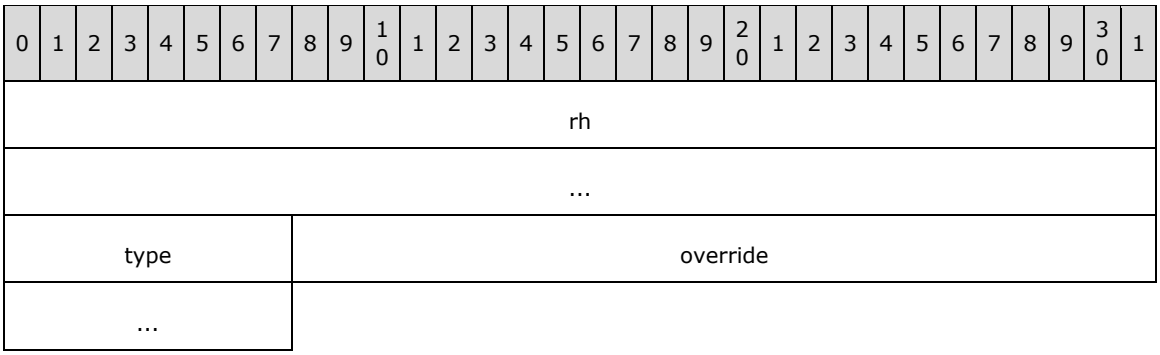
Value	Meaning
0x00000000	Use clockwise direction for the hue component in HSL color space.
0x00000001	Use counterclockwise direction for the hue component in HSL color space.

2.8.41 TimeOverride

Referenced by: TimeVariant4Behavior

An atom record that specifies how an animation behavior overrides the values of the properties being animated on an object.

Let the *corresponding time node* be as specified in the TimePropertyList2.8.37TimeBehavior record that contains this **TimeOverride** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be an RT_TimeVariant.
rh.recLen	MUST be 0x00000005.

type (1 byte): A **TimeVariantTypeEnum** enumeration that specifies the data type of this record. It MUST be TL_TVT_Int.

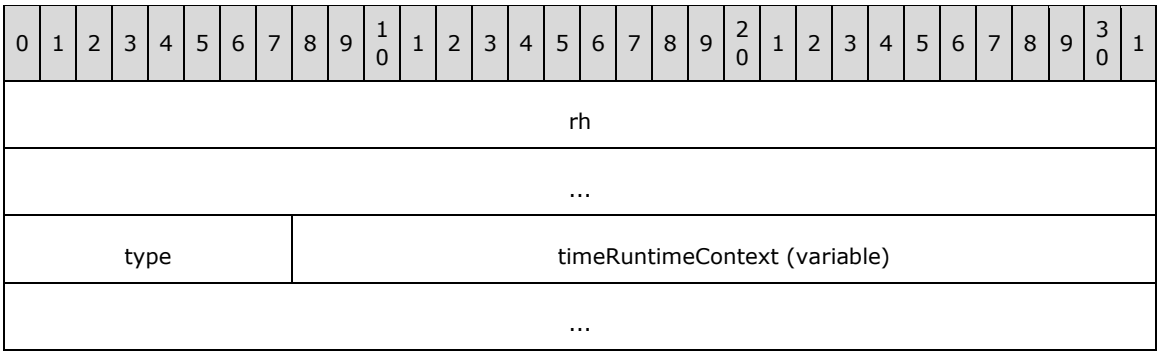
override (4 bytes): A TimeVariantInt record that specifies how to override the values of the properties of the target object being animated by the animation behavior of the *corresponding time node*. It MUST be 0x00000001 that specifies the animated properties of the sub-elements contained inside the target object are cleared and inherited from the target object before the animation.

2.8.42 TimeRuntimeContext

Referenced by: TimeVariant4Behavior

An atom record that specifies the runtime context of an animation behavior.

Let the *corresponding time node* be as specified in the TimePropertyList2.8.37TimeBehavior record that contains this **TimeRuntimeContext** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be an RT_TimeVariant.
rh.recLen	MUST be an odd number.

type (1 byte): A **TimeVariantTypeEnum** enumeration that specifies the data type of this record. It MUST be TL_TVT_String.

timeRuntimeContext (variable): A UnicodeString that specifies the runtime context of the animation behavior of the *corresponding time node*. It MUST be a valid RUNTIME_CONTEXT as specified in the following ABNF (specified in [RFC5234]) grammar:

```
RUNTIME_CONTEXT = CONTEXT_ATOM *(";" CONTEXT_ATOM) [";"]
CONTEXT_ATOM = [RELATION_OPERATOR SPACE] APP_ABBREV [SPACE VERSION]
RELATION_OPERATOR = GTE / GT / LTE / LT / NOT
APP_ABBREV =  ("p" / "P") ("p" / "P") ("t" / "T")

VERSION = DEC NUMBER ["." DEC NUMBER]
DEC_NUMBER = 1*DIGIT
GTE = "g" "t" "e"
GT = "g" "t"
LTE = "l" "t" "e"
LT = "l" "t"
NOT = "!"
SPACE = 1*" "
```

The length, in bytes, of the field is specified by the following formula:

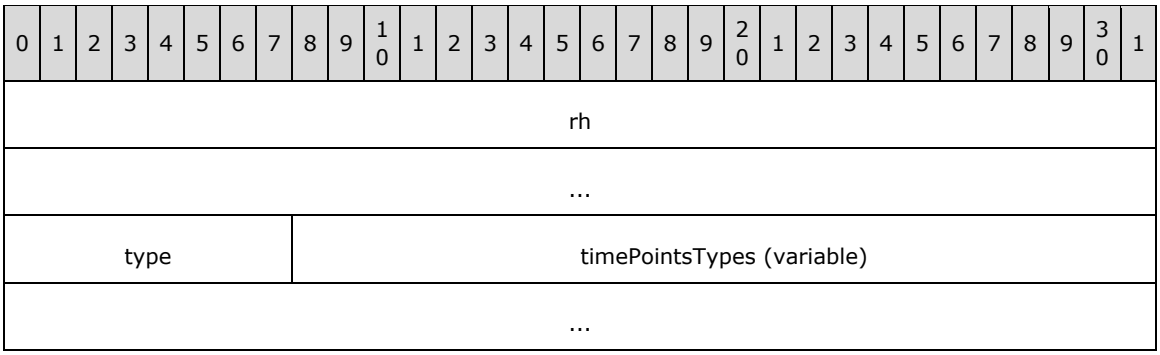
```
rh.recLen - 1
```

2.8.43 TimePointsTypes

Referenced by: TimeVariant4Behavior

An atom record that specifies the type of points in a motion path.

Let the *corresponding time node* be as specified in the TimePropertyList2.8.37TimeBehavior record that contains this **TimePointsTypes** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be an RT_TimeVariant.
rh.recLen	MUST be an odd number.

type (1 byte): A **TimeVariantTypeEnum** enumeration that specifies the data type of this record. It MUST be TL_TVT_String.

timePointsTypes (variable): A UnicodeString that specifies the type of points in the path attribute and the description of the motion path near the current point. The length, in bytes, of the field is specified by the following formula:

```
rh.recLen - 1
```

This field has no effect on the playing of the animation. It is only used when the motion path is edited in a user interface.

Each character in this string sequentially maps to a point defined in the path string as specified in the **varPath** field of the **TimeMotionBehaviorContainer** record (section 2.8.63). It MUST be a sequence formed with characters from the following table.

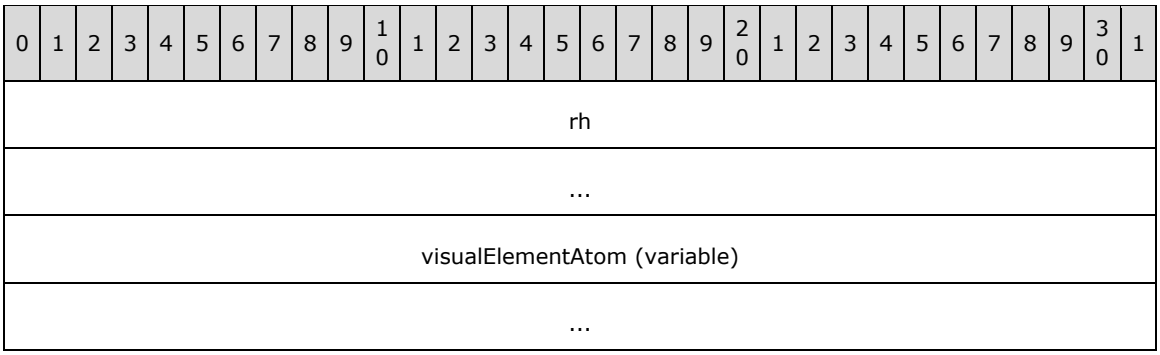
Character	Meaning
A	Auto line point.
a	Auto curve point.
F	Corner line point.
f	Corner curve point.
T	Straight line point.
t	Straight curve point.
S	Smooth line point.

Character	Meaning
s	Smooth curve point.

2.8.44 ClientVisualElementContainer

Referenced by: ExtTimeNodeContainer, SubEffectContainer, TimeBehaviorContainer, TimeConditionContainer

A container record that specifies the target for an animation effect.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Subfields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeClientVisualElement.

visualElementAtom (variable): A VisualElementAtom record that specifies the target to which the animation effect applies.

2.8.45 VisualElementAtom

Referenced by: ClientVisualElementContainer

A variable type record whose type and meaning are dictated by the value of **rh.recType**, as specified in the following table.

Value	Meaning
RT_VisualPageAtom	A VisualPageAtom record that specifies the slide as the target for a time condition.
RT_VisualShapeAtom	A VisualShapeOrSoundAtom record that specifies the shape or sound information for an animation target.

2.8.46 VisualPageAtom

Referenced by: VisualElementAtom

An atom record that specifies the slide as the target for a time condition.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
type																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_VisualPageAtom.
rh.recLen	MUST be 0x00000004.

type (4 bytes): A **TimeVisualElementEnum** enumeration that specifies the target element type. It MUST be TL_TVET_Page.

2.8.47 VisualShapeOrSoundAtom

Referenced by: VisualElementAtom

A variable type record whose type and meaning are dictated by the value of **refType**, as specified in the following table.

Value	Meaning
TL_ET_SoundType	A VisualSoundAtom record that specifies the sound information for an animation target.
TL_ET_ShapeType	A VisualShapeAtom record that specifies the shape information for an animation target.

2.8.48 VisualSoundAtom

Referenced by: VisualShapeOrSoundAtom

An atom record that specifies the sound information for an animation target.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
type																															
refType																															

soundIdRef
data1
data2

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_VisualShapeAtom.
rh.recLen	MUST be 0x00000014.

type (4 bytes): A **TimeVisualElementEnum** enumeration that specifies the target element type for the shape that the animation is applied to. It MUST NOT be TL_TVET_Page.

refType (4 bytes): An **ElementTypeEnum** enumeration that specifies the target element type of the animation. It MUST be TL_ET_SoundType.

soundIdRef (4 bytes): A SoundIdRef that specifies the value to look up in the **SoundCollectionContainer** record (section 2.4.16.1) to find the embedded audio.

data1 (4 bytes): MUST be 0xFFFFFFFF, and MUST be ignored.

data2 (4 bytes): MUST be 0xFFFFFFFF, and MUST be ignored.

2.8.49 VisualShapeAtom

Referenced by: VisualShapeOrSoundAtom

A variable type record whose type and meaning are dictated by the value of **type**, as specified in the following table.

Value	Meaning
TL_TVET_ChartElement	A VisualShapeChartElementAtom record that specifies the embedded chart shape for an animation target.
All other values	A VisualShapeGeneralAtom record that specifies the shape for an animation target. The shape MUST NOT be an embedded chart.

2.8.50 VisualShapeChartElementAtom

Referenced by: VisualShapeAtom

An atom record that specifies an embedded chart or its sub-elements to animate.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															

...
type
refType
shapeIdRef
data1
data2

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_VisualShapeAtom.
rh.recLen	MUST be 0x00000014.

type (4 bytes): A **TimeVisualElementEnum** enumeration that specifies the target element type in the shape to which the animation is applied. It MUST be TL_TVET_ChartElement.

refType (4 bytes): An **ElementTypeEnum** enumeration that specifies the target element type of the animation. It MUST be TL_ET_ShapeType.

shapeIdRef (4 bytes): An unsigned integer that specifies the target shape on the slide to animate.

data1 (4 bytes): An **unsigned integer** that specifies how the chart is built during its animation. It MUST be a value from the following table.

Value	Meaning
0x00000000	The entire chart.
0x00000001	By series.
0x00000002	By category.
0x00000003	By series element.
0x00000004	By category element.
0x00000005	Custom chart element.

data2 (4 bytes): A **signed integer** that specifies a chart element to animate. It MUST be greater than or equal to 0xFFFFFFFF (-1). The value 0xFFFFFFFF specifies that this record is invalid and SHOULD be ignored. The value 0x00000000 specifies the chart background. Values greater than 0x0000000 specify a one-based index in the list of chart elements specified by **data1**.

2.8.51 VisualShapeGeneralAtom

Referenced by: VisualShapeAtom

An atom record that specifies a shape or one of its parts as a target to animate.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
type																															
refType																															
shapeIdRef																															
data1																															
data2																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_VisualShapeAtom.
rh.recLen	MUST be 0x00000014.

type (4 bytes): A **TimeVisualElementEnum** enumeration that specifies the target element type in the shape to which the animation is applied. It MUST NOT be TL_TVET_Page.

refType (4 bytes): An **ElementTypeEnum** enumeration that specifies the target element type of the animation. It MUST be TL_ET_ShapeType.

shapeIdRef (4 bytes): An unsigned integer that specifies the target shape on the slide to animate.

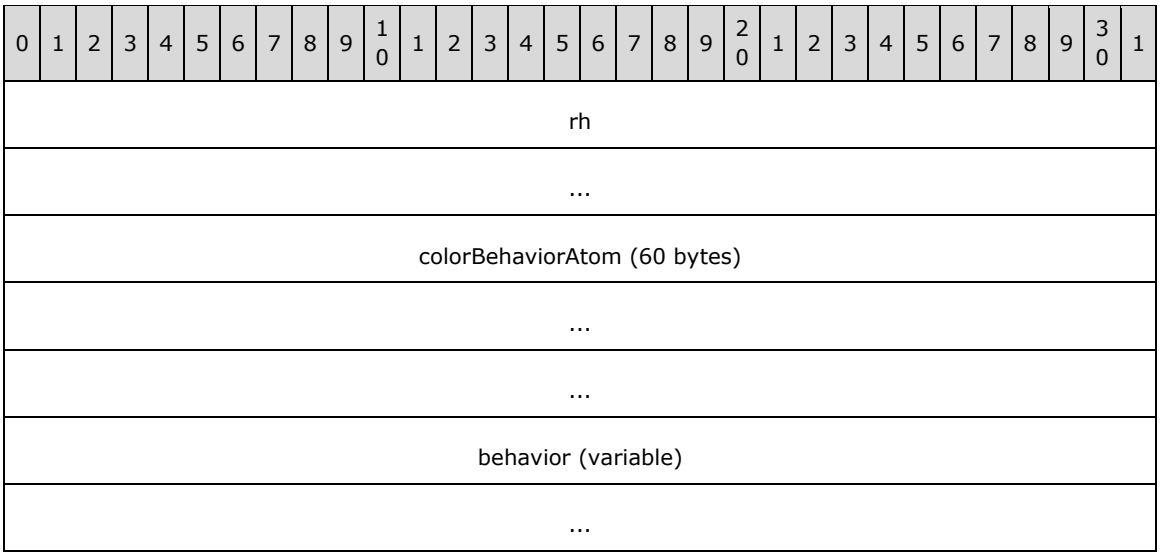
data1 (4 bytes): A signed integer that specifies the zero-based character index of the beginning of a text range. It MUST be ignored unless **type** is TL_TVET_TextRange.

data2 (4 bytes): A signed integer that specifies the zero-based character index of the end of a text range. It MUST be ignored unless **type** is TL_TVET_TextRange.

2.8.52 TimeColorBehaviorContainer

Referenced by: ExtTimeNodeContainer, SubEffectContainer

A container record that specifies a behavior that changes the color of an object. This animation behavior is applied to the object specified by the **behavior.clientVisualElement** field and used to animate one property specified by the **behavior.stringList** field. The property MUST be one from the following list that is a subset of the properties specified in the **TimeStringListContainer** record (section 2.8.36): "ppt_c", "style.color", "imageData.chromakey", "fill.color", "fill.color2", "stroke.color", "stroke.color2", "shadow.color", "shadow.color2", "extrusion.color", and "fillcolor".



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Subfields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeColorBehaviorContainer.

colorBehaviorAtom (60 bytes): A TimeColorBehaviorAtom record that specifies how to change the color of the object and which attributes within this field are valid.

behavior (variable): A **TimeBehaviorContainer** record (section 2.8.34) that specifies the common behavior information.

2.8.53 TimeColorBehaviorAtom

Referenced by: TimeColorBehaviorContainer

An atom record that specifies the information for an animation that changes the color of an object.

colorFrom (16 bytes)
...
...
colorTo (16 bytes)
...
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeColorBehavior.
rh.recLen	MUST be 0x00000034.

flag (4 bytes): A TimeColorBehaviorPropertyUsedFlag structure that specifies which fields are valid.

colorBy (16 bytes): A TimeAnimateColorBy structure that specifies the offset color value.

MUST be ignored if **colorTo** exists. It MUST be ignored if **flag.fByPropertyUsed** is **FALSE**.

colorFrom (16 bytes): A TimeAnimateColor structure that specifies the starting color value.

If **colorFrom** exists, **colorBy** or **colorTo** MUST also exist. It MUST be ignored if **flag.fFromPropertyUsed** is **FALSE**.

colorTo (16 bytes): A TimeAnimateColor structure that specifies the end color value.

MUST be ignored if **flag.fToPropertyUsed** is **FALSE**.

2.8.54 TimeColorBehaviorPropertyUsedFlag

Referenced by: TimeColorBehaviorAtom

A structure that specifies which attributes of a color animation are valid.

Let the *corresponding* ***TimeColorBehaviorAtom*** be specified by the TimeColorBehaviorAtom record that contains this **TimeColorBehaviorPropertyUsedFlag** record.

Let the *corresponding* ***TimeColorBehaviorContainer*** be specified by the **TimeColorBehaviorContainer** record (section 2.8.52) that contains the *corresponding* ***TimeColorBehaviorAtom***.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
A	B	C	D	E	reserved																										

A - fByPropertyUsed (1 bit): A bit that specifies whether the **colorBy** field of the corresponding **TimeColorBehaviorAtom** is valid.

B - fFromPropertyUsed (1 bit): A **bit** that specifies whether the **colorFrom** field of the *corresponding* ***TimeColorBehaviorAtom*** is valid.

C - fToPropertyUsed (1 bit): A **bit** that specifies whether the **colorTo** field of the corresponding **TimeColorBehaviorAtom** is valid.

D - fColorSpacePropertyUsed (1 bit): A **bit** that specifies whether the **behavior.propertyList.rec** field of the *corresponding* ***TimeColorBehaviorContainer*** which has **behavior.propertyList.rec.rh.recInstance** equal to TL_TBPID_ColorColorModel was explicitly set by a user interface action.

E - fDirectionPropertyUsed (1 bit): A **bit** that specifies whether the **behavior.propertyList.rec** field of the *corresponding* ***TimeColorBehaviorContainer*** which has **behavior.propertyList.rec.rh.recInstance** equal to TL_TBPID_ColorDirection was explicitly set by a user interface action.

reserved (27 bits): MUST be zero, and MUST be ignored.

2.8.55 TimeAnimateColorBy

Referenced by: TimeColorBehaviorAtom

A variable type structure whose type and meaning are dictated by the value of the **model** field within these three structures, as specified by the following table.

Value	Meaning
0x00000000	An RGBColorBy structure that specifies an RGB color offset.
0x00000001	An HSLColorBy structure that specifies an HSL color offset.
0x00000002	An IndexSchemeColor structure that specifies a color scheme color.

2.8.56 RGBColorBy

Referenced by: TimeAnimateColorBy

A structure that specifies the offset values during an animation of the red, green, and blue color components in RGB color space. These offset values are added to the starting value for each color component at specified time intervals until the animation is complete.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
model																															
red																															
green																															
blue																															

model (4 bytes): An unsigned integer that specifies this color is defined within the RGB color space. It MUST be 0x00000000.

red (4 bytes): A signed integer that specifies the offset value of the red color component. It MUST be greater than or equal to -255 and less than or equal to 255.

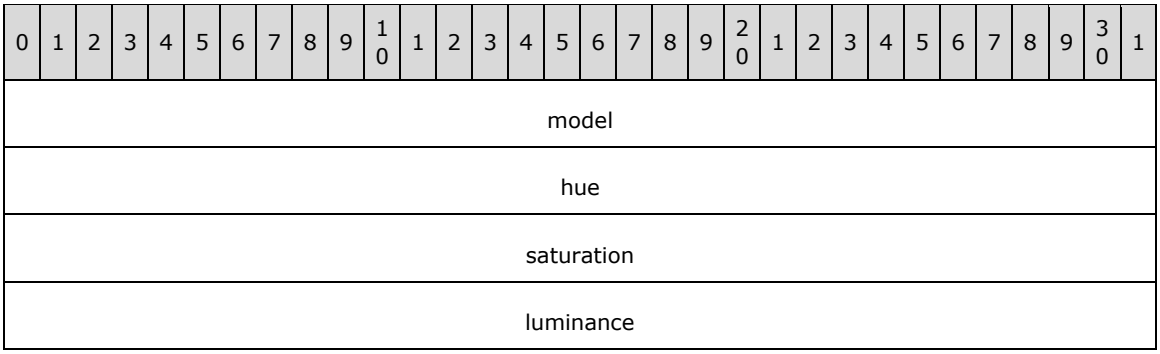
green (4 bytes): A signed integer that specifies the offset value of the green color component. It MUST be greater than or equal to -255 and less than or equal to 255.

blue (4 bytes): A signed integer that specifies the offset value of the blue color component. It MUST be greater than or equal to -255 and less than or equal to 255.

2.8.57 HSLColorBy

Referenced by: TimeAnimateColorBy

A structure that specifies the offset values during an animation of the hue, saturation, and luminance color components in HSL color space. These offset values are added to the starting value for each color component at specified time intervals until the animation is complete.



model (4 bytes): An unsigned integer that specifies that this color is defined within the HSL color space. It MUST be 0x00000001.

hue (4 bytes): A signed integer that specifies the offset value of the hue color component. It MUST be greater than or equal to -255 and less than or equal to 255.

saturation (4 bytes): A signed integer that specifies the offset value of the saturation color component. It MUST be greater than or equal to -255 and less than or equal to 255.

luminance (4 bytes): A signed integer that specifies the offset value of the luminance color component. It MUST be greater than or equal to -255 and less than or equal to 255.

2.8.58 TimeAnimateColor

Referenced by: TimeColorBehaviorAtom

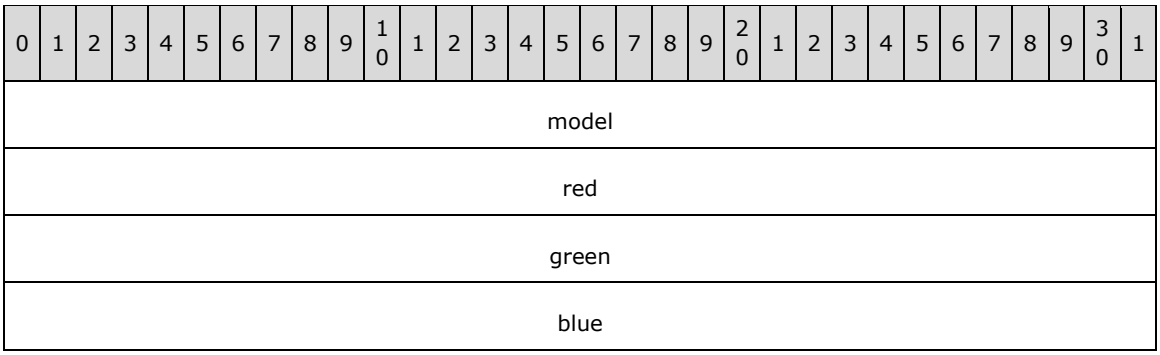
A variable type structure whose type and meaning are dictated by the value of the **model** field within these two structures, as specified by the following table.

Value	Meaning
0x00000000	An RGBColor structure that specifies an RGB color.
0x00000002	An IndexSchemeColor structure that specifies a color scheme color.

2.8.59 RGBColor

Referenced by: TimeAnimateColor

A structure that specifies the values of the red, green, and blue components of a color in the RGB color space.



model (4 bytes): An **unsigned integer** that specifies this color is defined within the RGB color space. It MUST be 0x00000000.

red (4 bytes): An **unsigned integer** that specifies the value of the red color component. It MUST be less than or equal to 255.

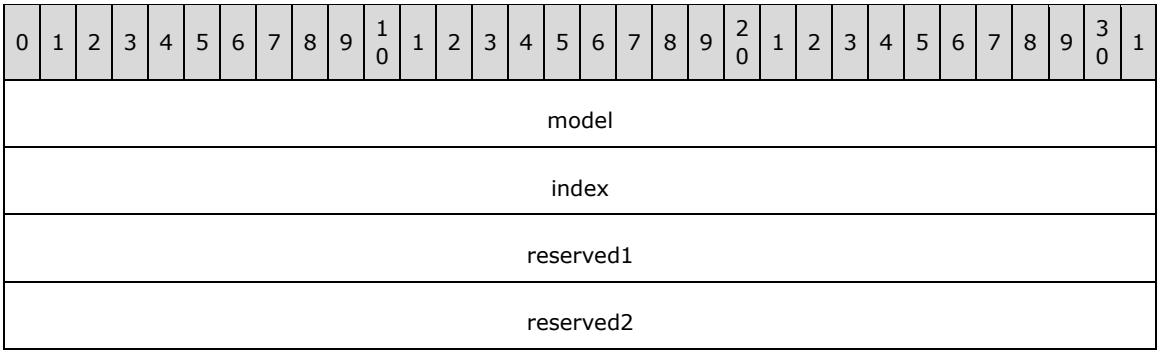
green (4 bytes): An **unsigned integer** that specifies the value of the green color component. It MUST be less than or equal to 255.

blue (4 bytes): An **unsigned integer** that specifies the value of the blue color component. It MUST be less than or equal to 255.

2.8.60 IndexSchemeColor

Referenced by: TimeAnimateColor, TimeAnimateColorBy

A structure that specifies a color from a color scheme.



model (4 bytes): An unsigned integer that specifies this color is from a color scheme. It MUST be 0x0000002.

index (4 bytes): An unsigned integer that specifies the index to the color scheme. It MUST be less than or equal to 7.

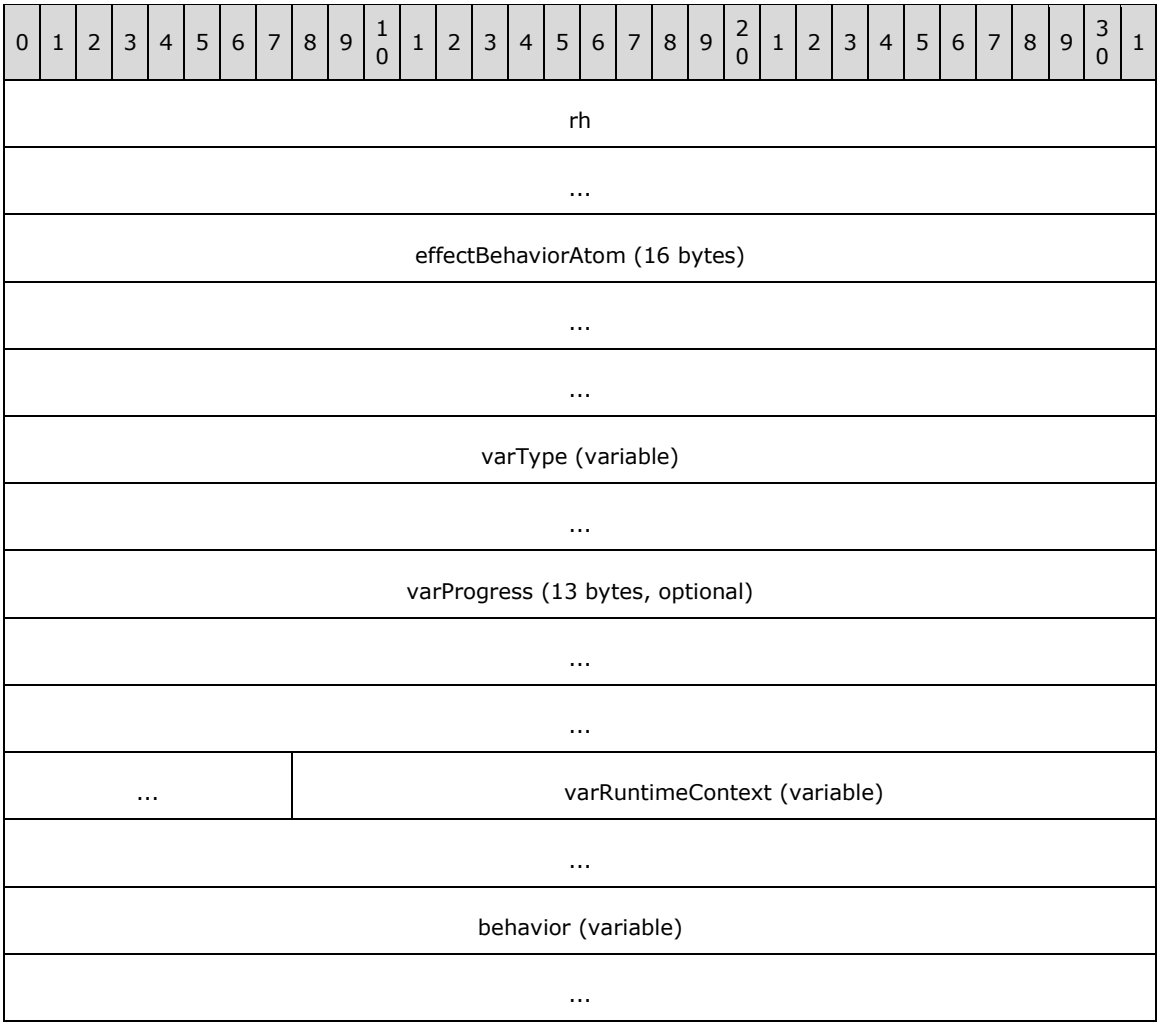
reserved1 (4 bytes): MUST be zero, and MUST be ignored.

reserved2 (4 bytes): MUST be zero, and MUST be ignored.

2.8.61 TimeEffectBehaviorContainer

Referenced by: ExtTimeNodeContainer

A container record that specifies an effect behavior that transforms the image of an object. The transformation provides the ability to perform transitions on objects. There is no property to be animated in this behavior. The **behavior.stringList** field is ignored.



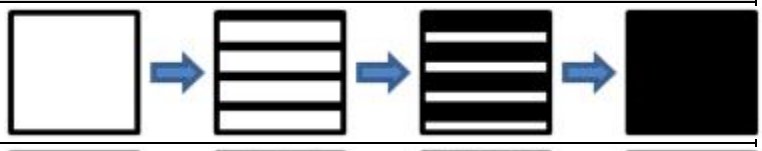
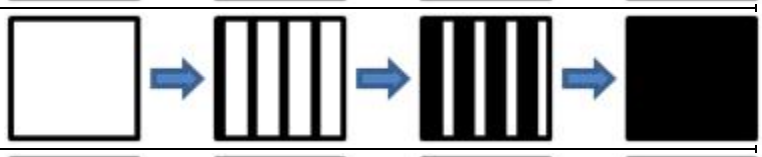
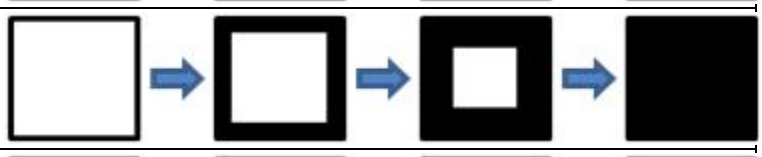
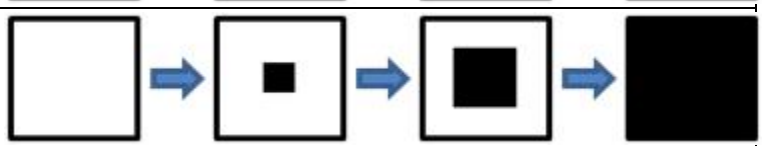
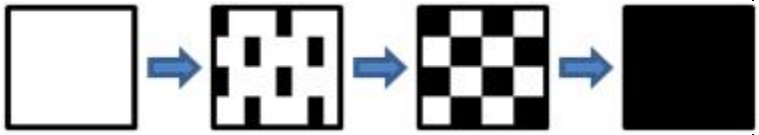
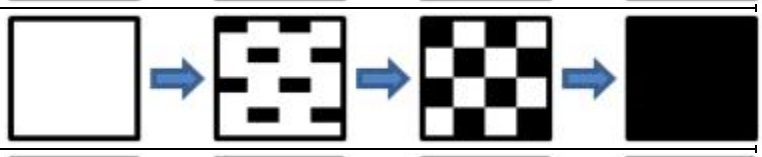
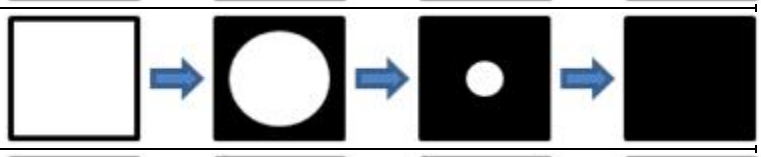
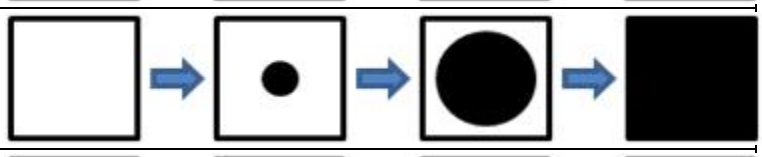
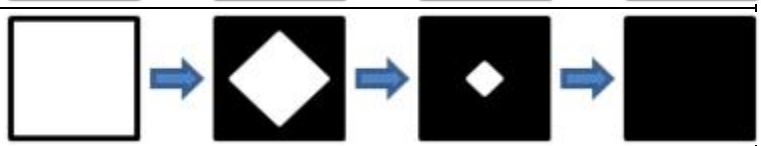
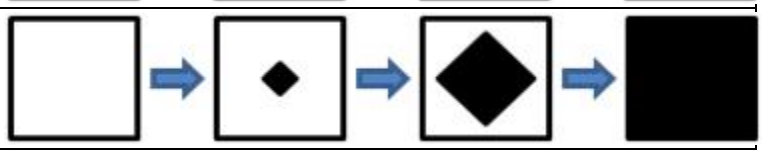
rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table:

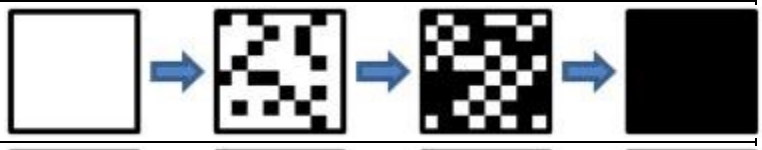
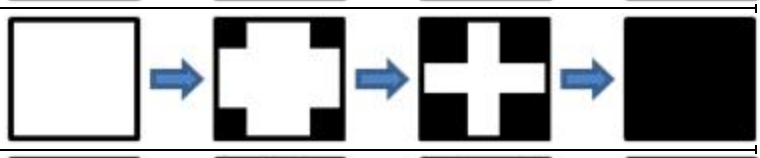
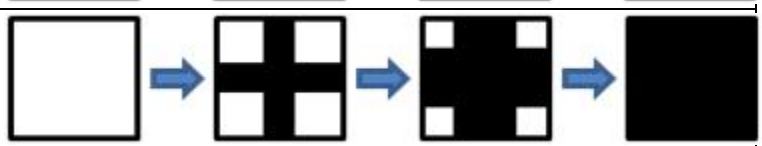
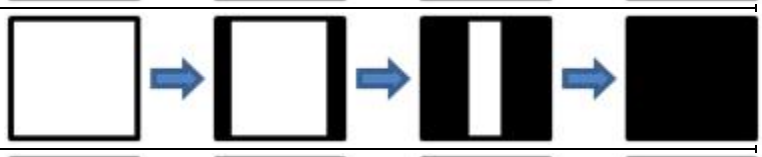
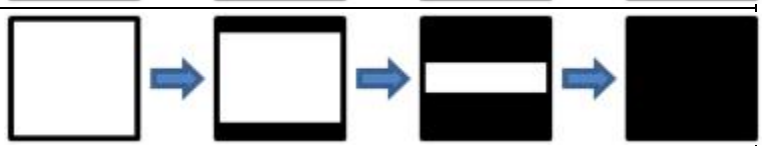
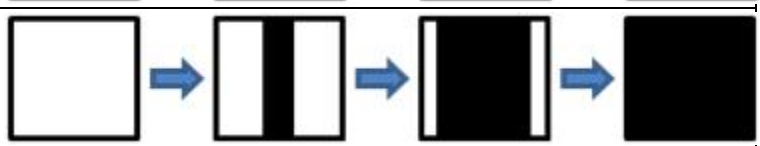
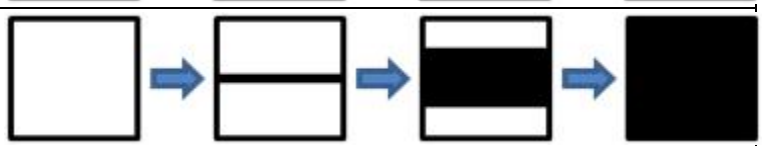
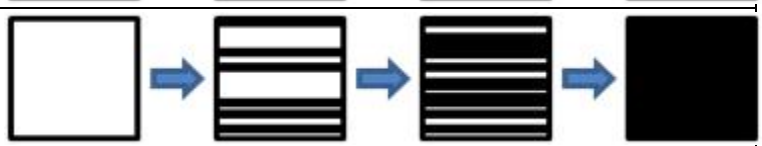
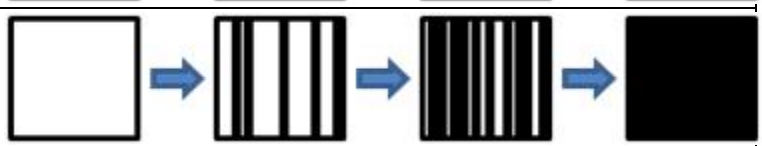
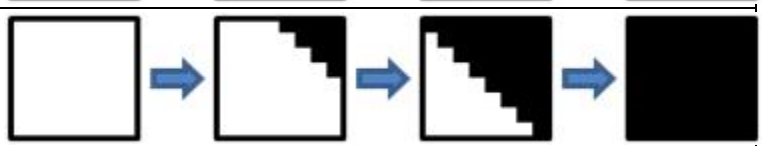
Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeEffectBehaviorContainer.

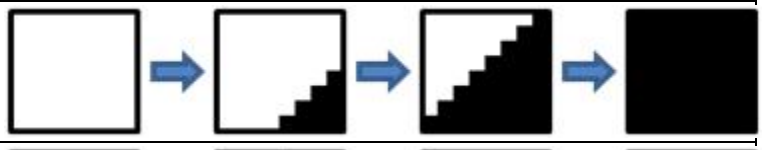
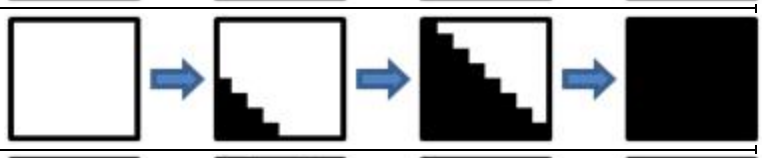
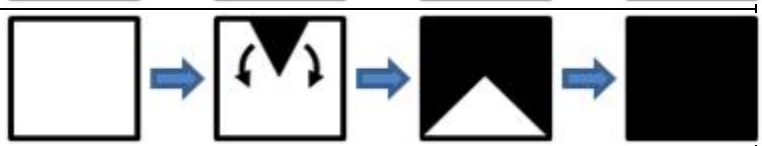
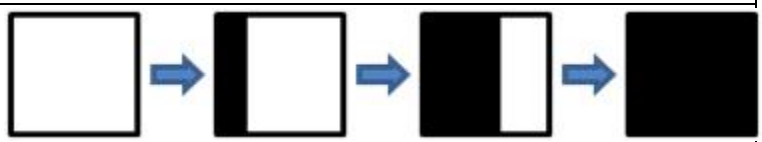
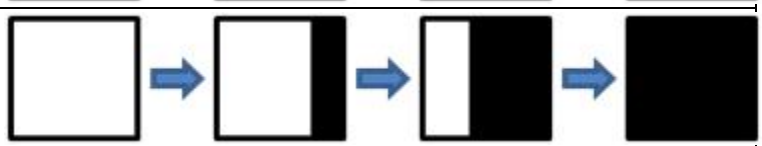
effectBehaviorAtom (16 bytes): A TimeEffectBehaviorAtom record that specifies which attributes of this field and this **TimeEffectBehaviorContainer** are valid, and the transformation style of the object.

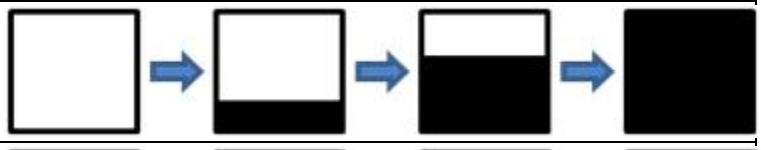
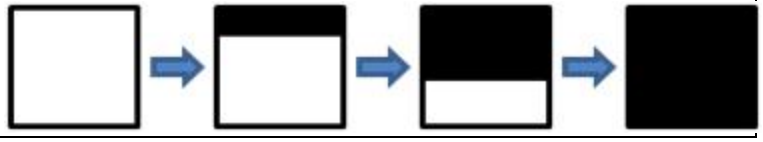
varType (variable): A TimeVariantString record that specifies the object transitions for the effect. The **varType.rh.recInstance** sub-field MUST be 0x001. It MUST be ignored if **effectBehaviorAtom.fTypePropertyUsed** is **FALSE**.

The **varType.stringValue** sub-field MUST be a value specified in the following table:

Value	Description
blinds(horizontal)	
blinds(vertical)	
box(in)	
box(out)	
checkerboard(across)	
checkerboard(down)	
circle(in)	
circle(out)	
diamond(in)	
diamond(out)	

Value	Description
dissolve	
fade	
plus(in)	
plus(out)	
barn(inVertical)	
barn(inHorizontal)	
barn(outVertical)	
barn(outHorizontal)	
randombar(horizontal)	
randombar(vertical)	
strips(downLeft)	

Value	Description
strips(upLeft)	
strips(downRight)	
strips(upRight)	
wedge	
wheel(1)	
wheel(2)	
wheel(3)	
wheel(4)	
wheel(8)	
wipe(right)	
wipe(left)	

Value	Description
wipe(up)	
wipe(down)	

varProgress (13 bytes): An optional TimeVariantFloat record that specifies the normalized time for which the state of the animation is displayed until the end time. It MUST be ignored if **effectBehaviorAtom.fProgressPropertyUsed** is **FALSE**. Sub-fields are further specified in the following table.

Field	Meaning
varProgress.rh.recInstance	MUST be 0x002.
varProgress.floatValue	MUST be greater than or equal to 0, the normalized start time of the animation, and less than or equal to 1, the normalized end time of the animation.

varRuntimeContext (variable): An optional TimeVariantString record that specifies the runtime context. It MUST be ignored if **effectBehaviorAtom.fRuntimeContextObsolete** is **FALSE**. Sub-fields are further specified in the following table.

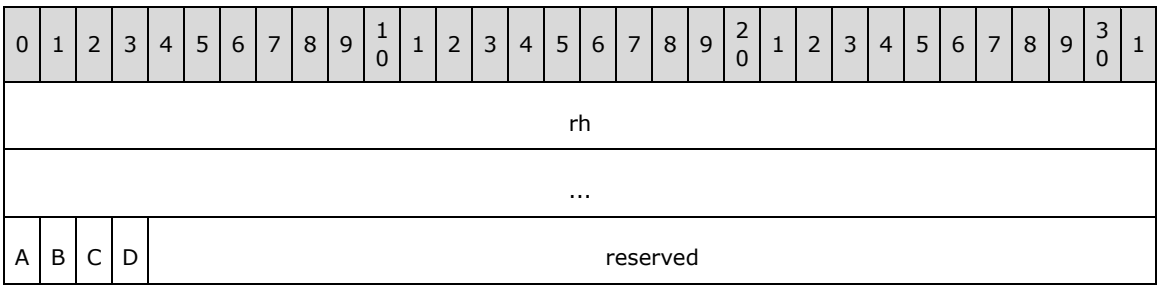
Field	Meaning
varRuntimeContext.rh.recInstance	MUST be 0x003.
varRuntimeContext.stringValue	MUST have a format as specified by the **timeRuntimeContext** field of the TimeRuntimeContext record

behavior (variable): A **TimeBehaviorContainer** record (section 2.8.34) that specifies the common behavior information.

2.8.62 TimeEffectBehaviorAtom

Referenced by: TimeEffectBehaviorContainer

An atom record that specifies the information of an animation that transforms the image of an object.



effectTransition

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeEffectBehavior.
rh.recLen	MUST be 0x00000008.

A - fTransitionPropertyUsed (1 bit): A **bit** that specifies whether **effectTransition** was explicitly set by a user interface action.

B - fTypePropertyUsed (1 bit): A bit that specifies whether the **varType** of the **TimeEffectBehaviorContainer** record (section 2.8.61) that contains this **TimeEffectBehaviorAtom** is valid.

C - fProgressPropertyUsed (1 bit): A bit that specifies whether the **varProgress** of the **TimeEffectBehaviorContainer** record that contains this **TimeEffectBehaviorAtom** is valid.

D - fRuntimeContextObsolete (1 bit): A bit that specifies whether the **varRuntimeContext** of the **TimeEffectBehaviorContainer** record that contains this **TimeEffectBehaviorAtom** is valid.

reserved (28 bits): MUST be zero, and MUST be ignored.

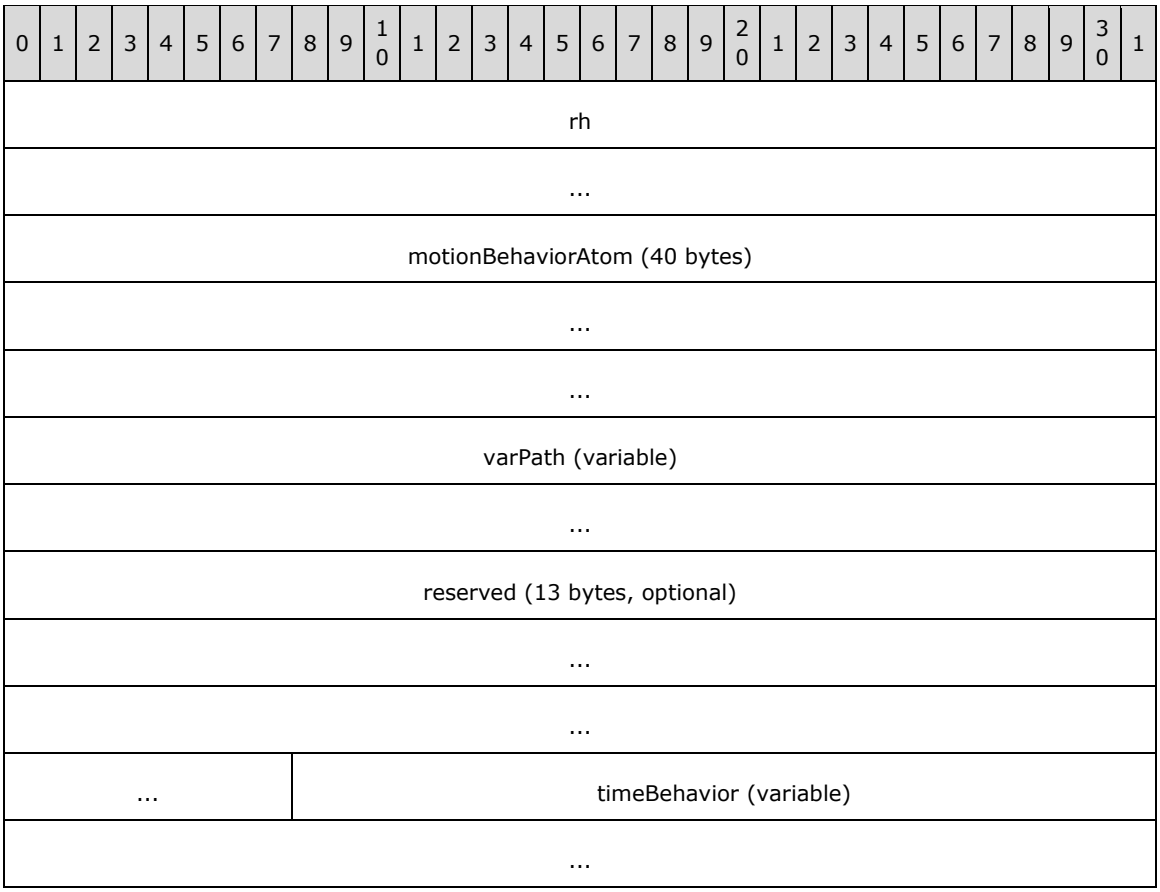
effectTransition (4 bytes): An unsigned integer that specifies how the image of the object is transformed. It MUST be ignored if **fTransitionPropertyUsed** is **FALSE** and a value of 0x00000000 MUST be used instead. It MUST be a value from the following table.

Value	Meaning
0x00000000	Transition in: the object is completely invisible at the beginning of the transformation and becomes completely visible at the end of the transformation.
0x00000001	Transition out: the object is completely visible at the beginning of the transformation and becomes completely invisible at the end of the transformation.

2.8.63 TimeMotionBehaviorContainer

Referenced by: ExtTimeNodeContainer

A container record that specifies a motion behavior that moves an object along a path. This animation behavior is applied to the object specified by the **timeBehavior.clientVisualElement** field and used to animate two properties specified by the **timeBehavior.stringList** field. The properties MUST be ones from the list that is specified in the **TimeStringListContainer** record (section 2.8.36). If no properties are specified, "ppt_x" and "ppt_y" will be used. If only one property is specified, "ppt_y" will be used as the second property.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeMotionBehaviorContainer.

motionBehaviorAtom (40 bytes): A TimeMotionBehaviorAtom record that specifies the origin of the path, which attributes of this field and this **TimeMotionBehaviorContainer** record are valid, and a path specified by a starting point, an endpoint, and an offset value.

varPath (variable): An optional TimeVariantString record that specifies the path for the animation motion. The **varPath.rh.recInstance** sub-field MUST be 0x001. It MUST be a valid MOTION_PATH as specified in the following ABNF (specified in [RFC5234]) grammar:

```
MOTION_PATH = 1*PATH_COMMAND
PATH_COMMAND = MOVE_COMMAND / LINE_COMMAND / CURVE_COMMAND / CLOSE_COMMAND
MOVE_COMMAND = ("m" / "M") SPACE COORD_NUMBER SPACE COORD_NUMBER
LINE_COMMAND = ("l" / "L") SPACE COORD_NUMBER SPACE COORD_NUMBER
CURVE_COMMAND = ("c" / "C") SPACE COORD NUMBER SPACE COORD NUMBER SPACE COORD NUMBER SPACE
COORD_NUMBER SPACE COORD_NUMBER SPACE COORD_NUMBER
CLOSE_COMMAND = ("z" / "Z")
COORD_NUMBER = (DEC_NUMBER ["." DEC_NUMBER]) / ("(" FORMULA ")")
DEC_NUMBER = 1*DIGIT
```

```
SPACE = 1*" "
```

The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.

The path is specified by defining an action and coordinates that go along with the action. The allowed action types that are understood within a path are listed in the following table. If the action is expressed in uppercase, the following point(s) are to be interpreted as absolute coordinates, or a point on the slide. If the action is expressed in lowercase, the point(s) are to be interpreted as relative coordinates, or an offset from the current position.

Action	Meaning
M = move to	This action requires a point specified by two coordinates. Move to will move the object to the specified point. It does not animate the object to that point; rather, the object will snap to the given point.
L = line to	This action requires a point specified by two coordinates. Line to will move the object to the specified point along the shortest line between the current point and the specified point.
C = curve to	This action requires three points specified by six coordinates. Curve to will move the object along a cubic Bezier curve specified by the current point and the three provided points.
Z=close loop	This action requires no points. Close loop will move the object back to where it was before the path began along the shortest line between the current point and the starting point. The relative and absolute versions of this action are identical.
E=end	This action requires no points. End will terminate the motion path. Any action specified in the string after the End action is ignored. If this action is not present at the end of a string and the string ends, this action will be implied. The relative and absolute versions of this action are identical.

Thus the total allowed set is {M,L,C,Z,E,m,l,c,z,e).

Points are expressed as normalized values of the slide size, for example, 1,1 means the lower-right corner of the slide in absolute coordinates, or the slide width and height in relative coordinates. Expressing a coordinate less than 1 but greater than 0 must be prefixed with 0 before the decimal point.

Formulas can also be used for any coordinate. To use a formula, the entire formula must be inside parentheses. Formula syntax is specified in the **varFormula** field of the TimeAnimationValueListEntry record.

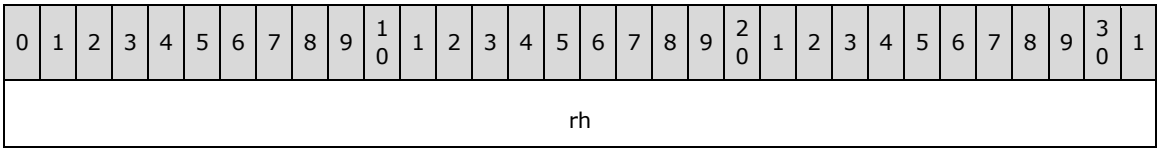
reserved (13 bytes): An optional TimeVariantInt record that MUST be ignored.

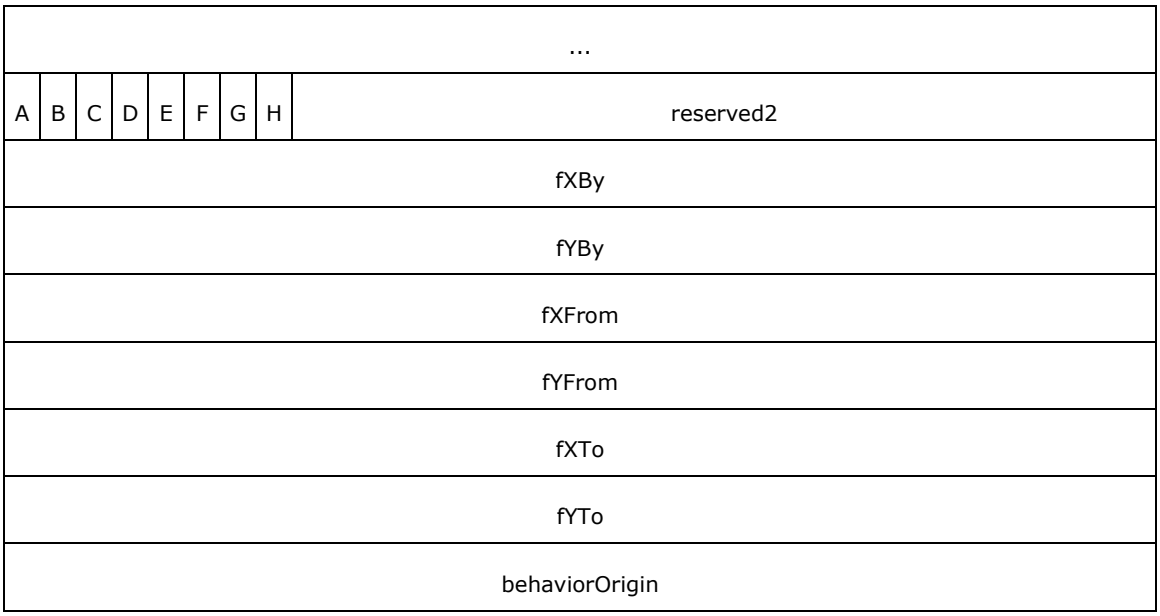
timeBehavior (variable): A **TimeBehaviorContainer** record (section 2.8.34) that specifies the common behavior information.

2.8.64 TimeMotionBehaviorAtom

Referenced by: TimeMotionBehaviorContainer

An atom record that specifies the information of an animation that moves an object along a path.





rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeMotionBehavior.
rh.recLen	MUST be 0x00000020.

A - fByPropertyUsed (1 bit): A bit that specifies whether **fXBy** and **fYBy** were explicitly set by a user interface action.

B - fFromPropertyUsed (1 bit): A bit that specifies whether **fXFrom** and **fYFrom** were explicitly set by a user interface action If **fFromPropertyUsed** is **TRUE**, **fByPropertyUsed** or **fToPropertyUsed** MUST also be **TRUE**.

C - fToPropertyUsed (1 bit): A bit that specifies whether **fXTo** and **fYTo** were explicitly set by a user interface action.

D - fOriginPropertyUsed (1 bit): A bit that specifies whether **behaviorOrigin** was explicitly set by a user interface action.

E - fPathPropertyUsed (1 bit): A bit that specifies whether the **varPath** of the **TimeMotionBehaviorContainer** record (section 2.8.63) that contains this **TimeMotionBehaviorAtom** is valid.

F - reserved1 (1 bit): MUST be 0x0.

G - fEditRotationPropertyUsed (1 bit): A bit that specifies whether the **timeContainer.propertyList** of the **TimeMotionBehaviorContainer** record that contains this **TimeMotionBehaviorAtom** has items of types TL_TBPID_PathEditRotationAngle, TL_TBPID_PathEditRotationX, and TL_TBPID_PathEditRotationY as specified in the **TimePropertyID2.13.37TimeBehavior** enumeration.

H - fPointsTypesPropertyUsed (1 bit): A bit that specifies whether the **timeContainer.propertyList** of the **TimeMotionBehaviorContainer** record that contains this **TimeMotionBehaviorAtom** has items of type TL_TBPID_PointsTypes as specified in the **TimePropertyID2.13.37TimeBehavior** enumeration.

reserved2 (24 bits): MUST be 0x000000.

fXBy (4 bytes): A floating-point number that specifies the offset value of the object position in the horizontal axis. It MUST be ignored if **fByPropertyUsed** is **FALSE** or if **fToPropertyUsed** is **TRUE**.

fYBy (4 bytes): A floating-point number that specifies the offset value of the object position in the vertical axis. It MUST be ignored if **fByPropertyUsed** is **FALSE** or if **fToPropertyUsed** is **TRUE**.

fXFrom (4 bytes): A floating-point number that specifies the starting position of the object in the horizontal axis. It MUST be ignored if **fFromPropertyUsed** is **FALSE** and a value of 0x00000000 MUST be used instead.

fYFrom (4 bytes): A floating-point number that specifies the starting position of the object in the vertical axis. It MUST be ignored if **fFromPropertyUsed** is **FALSE** and a value of 0x00000000 MUST be used instead.

fXTo (4 bytes): A floating-point number that specifies the end position of the object in the horizontal axis. It MUST be ignored if **fToPropertyUsed** is **FALSE**.

fYTo (4 bytes): A floating-point number that specifies the end position of the object in the vertical axis. It MUST be ignored if **fToPropertyUsed** is **FALSE**.

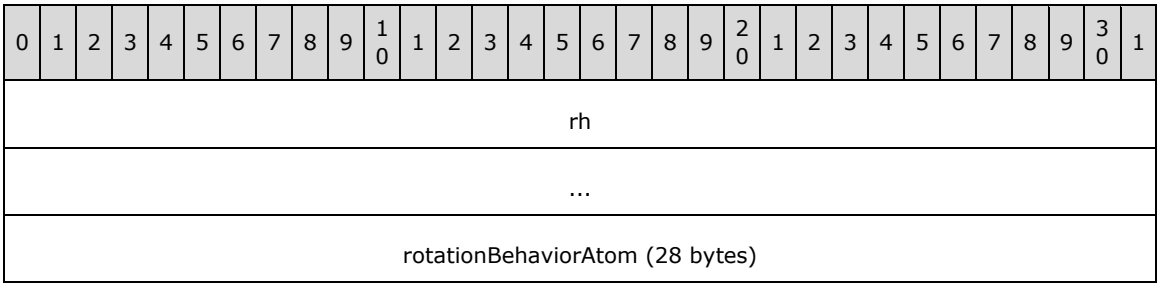
behaviorOrigin (4 bytes): An **unsigned integer** that specifies the origin of the motion path. It MUST be ignored if **fOriginPropertyUsed** is **FALSE** and a value of 0x00000002 MUST be used instead. It MUST be a value from the following table.

Value	Meaning
0x00000000	The origin is at the upper left corner of the slide that contains the object.
0x00000001	Same as 0x00000000.
0x00000002	The origin is at the center of the object.

2.8.65 TimeRotationBehaviorContainer

Referenced by: ExtTimeNodeContainer

A container record that specifies a rotation behavior that rotates an object. This animation behavior is applied to the object specified by the **behavior.clientVisualElement** field and used to animate one property specified by the **behavior.stringList** field. The property MUST be "r" or "ppt_r" from the list that is specified in the **TimeStringListContainer** record (section 2.8.36).



...
...
behavior (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeRotationBehaviorContainer.

rotationBehaviorAtom (28 bytes): A TimeRotationBehaviorAtom record that specifies how to rotate the object and which attributes within this field are valid.

behavior (variable): A **TimeBehaviorContainer** record (section 2.8.34) that specifies the common behavior information.

2.8.66 TimeRotationBehaviorAtom

Referenced by: TimeRotationBehaviorContainer

An atom record that specifies animation information for object rotation.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
A	B	C	D	reserved																											
fBy																															
fFrom																															
fTo																															
rotationDirection																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.

Field	Meaning
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeRotationBehavior.
rh.recLen	MUST be 0x00000014.

A - fByPropertyUsed (1 bit): A bit that specifies whether **fBy** was explicitly set by a user interface action.

B - fFromPropertyUsed (1 bit): A bit that specifies whether **fFrom** was explicitly set by a user interface action. If **fFromPropertyUsed** is **TRUE**, **fByPropertyUsed** or **fToPropertyUsed** MUST also be **TRUE**.

C - fToPropertyUsed (1 bit): A bit that specifies whether **fTo** was explicitly set by a user interface action.

D - fDirectionPropertyUsed (1 bit): A **bit** that specifies whether **rotationDirection** was explicitly set by a user interface action.

reserved (28 bits): MUST be zero, and MUST be ignored.

fBy (4 bytes): A floating-point number that specifies the offset degree of rotation. It MUST be ignored if **fByPropertyUsed** is **FALSE** or if **fToPropertyUsed** is **TRUE**.

fFrom (4 bytes): A floating-point number that specifies the starting degree of rotation. It MUST be ignored if **fFromPropertyUsed** is **FALSE** and a value of 0 MUST be used instead.

fTo (4 bytes): A floating-point number that specifies the end degree of rotation. It MUST be ignored if **fToPropertyUsed** is **FALSE** and a value of 360 MUST be used instead.

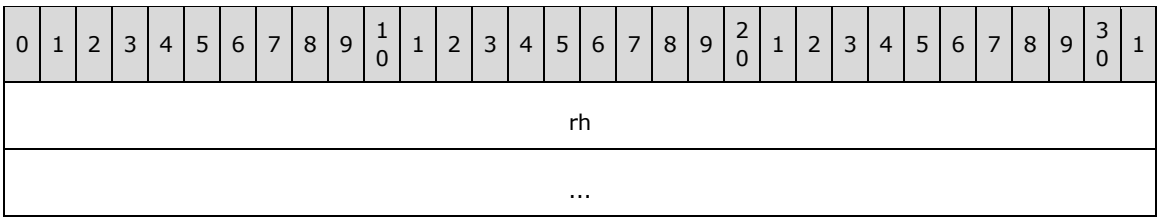
rotationDirection (4 bytes): An **unsigned integer** that specifies the rotation direction. It MUST be ignored if **fDirectionPropertyUsed** is **FALSE** and a value of 0x00000000 MUST be used instead. It MUST be a value from the following table.

Value	Meaning
0x00000000	Rotate clockwise.
0x00000001	Rotate counter clockwise.

2.8.67 TimeScaleBehaviorContainer

Referenced by: ExtTimeNodeContainer

A container record that specifies a scale behavior that animates the size of an object. This animation behavior is applied to the object specified by the **behavior.clientVisualElement** field and used to animate two properties including "ScaleX" and "ScaleY" from the list that is specified in the **TimeStringListContainer** record (section 2.8.36). The **behavior.stringList** field is ignored.



scaleBehaviorAtom (40 bytes)
...
...
behavior (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeScaleBehaviorContainer.

scaleBehaviorAtom (40 bytes): A TimeScaleBehaviorAtom record that specifies how to scale the size of an object and which attributes within this field are valid.

behavior (variable): A **TimeBehaviorContainer** record (section 2.8.34) that specifies the common behavior information.

2.8.68 TimeScaleBehaviorAtom

Referenced by: TimeScaleBehaviorContainer

An atom record that specifies animation information for scaling the size of an object.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
A	B	C	D	reserved																											
fXBy																															
fYBy																															
fXFrom																															
fYFrom																															
fXTo																															
fYTo																															

fZoomContents	unused

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeScaleBehavior.
rh.recLen	MUST be 0x00000020.

A - fByPropertyUsed (1 bit): A bit that specifies whether **fXBy** and **fYBy** were explicitly set by a user interface action.

B - fFromPropertyUsed (1 bit): A bit that specifies whether **fXFrom** and **fYFrom** were explicitly set by a user interface action. If **fFromPropertyUsed** is **TRUE**, **fByPropertyUsed** or **fToPropertyUsed** MUST also be **TRUE**.

C - fToPropertyUsed (1 bit): A bit that specifies whether **fXTo** and **fYTo** were explicitly set by a user interface action.

D - fZoomContentsUsed (1 bit): A bit that specifies whether **fZoomContents** was explicitly set by a user interface action.

reserved (28 bits): MUST be zero, and MUST be ignored.

fXBy (4 bytes): A floating-point number that specifies the offset value of the width of the object that is animated. It MUST be ignored if **fByPropertyUsed** is **FALSE** or if **fToPropertyUsed** is **TRUE.**

fYBy (4 bytes): A floating-point number that specifies the offset value of the height of the object that is animated. It MUST be ignored if **fByPropertyUsed** is **FALSE** or if **fToPropertyUsed** is **TRUE.**

fXFrom (4 bytes): A floating-point number that specifies the starting value of the width of the object that is animated. It MUST be ignored if **fFromPropertyUsed** is **FALSE** and a value of 0 MUST be used instead.

fYFrom (4 bytes): A floating-point number that specifies the starting value of the height of the object that is animated. It MUST be ignored if **fFromPropertyUsed** is **FALSE** and a value of 0 MUST be used instead.

fXTo (4 bytes): A floating-point number that specifies the end value of the width of the object that is animated. It MUST be ignored if **fToPropertyUsed** is **FALSE** and a value of 100 MUST be used instead.

fYTo (4 bytes): A floating-point number that specifies the end value of the height of the object that is animated. It MUST be ignored if **fToPropertyUsed** is **FALSE** and a value of 100 MUST be used instead.

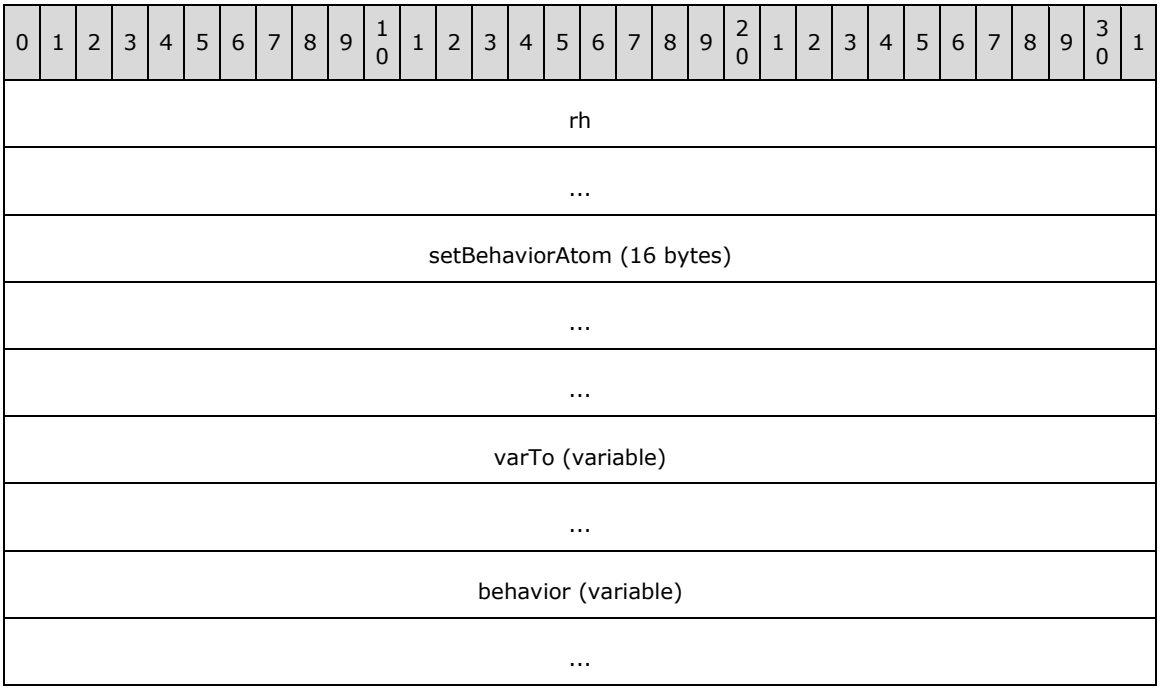
fZoomContents (1 byte): A **bool1** (section 2.2.2) that specifies whether the content contained by the scaling object is also scaled. It MUST be ignored if **fZoomContentsUsed** is **FALSE** and a value of 0x01 MUST be used instead.

unused (3 bytes): Undefined and MUST be ignored.

2.8.69 TimeSetBehaviorContainer

Referenced by: ExtTimeNodeContainer, SubEffectContainer

A container record that specifies a set behavior that assigns a value to a property. This animation behavior is applied to the object specified by the **behavior.clientVisualElement** field and used to animate one property specified by the **behavior.stringList** field. The property MUST be from the list that is specified in the **TimeStringListContainer** record (section 2.8.36).



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Subfields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeSetBehaviorContainer.

setBehaviorAtom (16 bytes): A TimeSetBehaviorAtom record that specifies the type of the value to be set and which attributes of this field and this **TimeSetBehaviorContainer** are valid.

varTo (variable): A TimeVariantString record that specifies the value that to be set to the property. The **varTo.rh.recInstance** subfield MUST be 0x001. It MUST be ignored if **setBehaviorAtom.fToPropertyUsed** is FALSE.

The allowed set of strings depends on the specific attribute names used in the **behavior.stringList** field. The supported attribute names are specified by the **TimeStringListContainer** record. The **varTo.stringValue** MUST be a value as specified in the following tables.

When using attribute names in the following table, the **setBehaviorAtom.valueType** field MUST be TL_TABVT_Number.

Attribute Name	varTo.stringValue Preset
style.visibility	"hidden", "visible"
style.fontWeight	"none", "normal", "bold"
style.fontStyle	"none", "normal", "italic"
style.textEffectEmboss	"none", "normal", "emboss"
style.textShadow	"none", "normal", "auto"
style.textTransform	"none", "normal", "sub", "super"
style.textDecorationUnderline	"false", "true"
style.textEffectOutline	"false", "true"
style.textDecorationLineThrough	"false", "true"
imageData.grayscale	"false", "true"
fill.on	"false", "f", "t", "true"
fill.type	"solid", "pattern", "tile", "frame", "gradientUnscaled", "gradient", "gradientCenter", "gradientRadial", "gradientTile", "background"
fill.method	"none", "linear", "sigma", "any"
stroke.on	"false", "f", "t", "true"
stroke.linestyle	"single", "thinThin", "thinThick", "thickThin", "thickBetweenThin"
stroke.dashstyle	"solid", "dot", "dash", "dashDot", "longDash", "longDashDot", "longDashDotDot"
stroke.filltype	"solid", "tile", "pattern", "frame"
stroke.startArrow	"none", "block", "classic", "diamond", "oval", "open", "chevron", "doublechevron"
stroke.endArrow	"none", "block", "classic", "diamond", "oval", "open", "chevron", "doublechevron"
stroke.startArrowWidth	"narrow", "medium", "wide"
stroke.startArrowLength	"short", "medium", "long"
stroke.endArrowWidth	"narrow", "medium", "wide"
stroke.endArrowLength	"short", "medium", "long"
shadow.on	"false", "f", "t", "true"
shadow.type	"single", "double", "emboss", "perspective"
skew.on	"false", "f", "t", "true"
extrusion.on	"false", "f", "t", "true"
extrusion.type	"parallel", "perspective"
extrusion.render	"solid", "wireframe", "boundingcube"
extrusion.plane	"xy", "zx", "yz"
extrusion.lockrotationcenter	"false", "true"
extrusion.autorotationcenter	"false", "true"
extrusion.colormode	"false", "true"

When using attribute names in the following table, the **setBehaviorAtom.valueType** field MUST be TL_TABVT_Number.

Attribute Name	varTo.stringValue
ppt_x	MUST be a valid FORMULA_OR_NUMBER as specified in the following **ABNF** (specified in [RFC5234]) grammar: `FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER`

Attribute Name	varTo.stringValue
	SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC REGULAR VALUE = ["-"] DEC NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
ppt_y	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
ppt_w	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
ppt_h	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC REGULAR VALUE = ["-"] DEC NUMBER ["."] [DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
ppt_r	MUST be a valid FORMULA_OR_NUMBER as specified in the following

Attribute Name	varTo.stringValue
	ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC REGULAR VALUE = ["-"] DEC NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
xshear	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC REGULAR VALUE = ["-"] DEC NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
yshear	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
ScaleX	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC EXPONENT] DEC_NUMBER = 1*DIGIT

Attribute Name	varTo.stringValue
	DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
ScaleY	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
r	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
style.opacity	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
style.rotation	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."]

Attribute Name	varTo.stringValue
	[DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
style.fontSize	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC NUMBER] [DEC EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
style.sRotation	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
imageData.cropTop	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
imageData.cropBottom	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar:

Attribute Name	varTo.stringValue
	FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC REGULAR VALUE = ["-"] DEC NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
imageData.cropLeft	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC REGULAR VALUE = ["-"] DEC NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
imageData.cropRight	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
imageData.gain	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC REGULAR VALUE = ["-"] DEC NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the

Attribute Name	varTo.stringValue
	TimeAnimationValueListEntry record.
imageData.blacklevel	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
imageData.gamma	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
fill.opacity	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
fill.opacity2	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC_NUMBER = 1*DIGIT

Attribute Name	varTo.stringValue
	DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
fill.angle	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
fill.focus	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
fill.focusposition.x	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
fill.focusposition.y	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."]

Attribute Name	varTo.stringValue
	[DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
fill.focussize.x	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC NUMBER] [DEC EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
fill.focussize.y	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
stroke.weight	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
stroke.opacity	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar:

Attribute Name	varTo.stringValue
	FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC REGULAR VALUE = ["-"] DEC NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
stroke.imagesize.x	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC REGULAR VALUE = ["-"] DEC NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
stroke.imagesize.y	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
shadow.opacity	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC REGULAR VALUE = ["-"] DEC NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the

Attribute Name	varTo.stringValue
	TimeAnimationValueListEntry record.
shadow.offset.x	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
shadow.offset.y	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
shadow.offset2.x	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
shadow.offset2.y	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC_NUMBER = 1*DIGIT

Attribute Name	varTo.stringValue
	DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
shadow.origin.x	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
shadow.origin.y	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
shadow.matrix.xtox	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
shadow.matrix.ytox	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."]

Attribute Name	varTo.stringValue
	[DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
shadow.matrix.ytoy	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC NUMBER] [DEC EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
shadow.matrix.perspectiveX	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
shadow.matrix.perspectiveY	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
skew.offset.x	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar:

Attribute Name	varTo.stringValue
	FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC REGULAR VALUE = ["-"] DEC NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
skew.offset.y	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC REGULAR VALUE = ["-"] DEC NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
skew.origin.x	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
skew.origin.y	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC REGULAR VALUE = ["-"] DEC NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the

Attribute Name	varTo.stringValue
	TimeAnimationValueListEntry record.
skew.matrix.xtox	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
skew.matrix.ytox	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
skew.matrix.ytoy	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
skew.matrix.perspectiveX	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC_NUMBER = 1*DIGIT

Attribute Name	varTo.stringValue
	DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
skew.matrix.perspectiveY	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
extrusion.viewpointorigin.x	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
extrusion.viewpointorigin.y	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
extrusion.viewpoint.x	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."]

Attribute Name	varTo.stringValue
	[DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
extrusion.viewpoint.y	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC NUMBER] [DEC EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
extrusion.viewpoint.z	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
extrusion.skewangle	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
extrusion.skewamt	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar:

Attribute Name	varTo.stringValue
	FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC REGULAR VALUE = ["-"] DEC NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
extrusion.backdepth	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC REGULAR VALUE = ["-"] DEC NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
extrusion.foredepth	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
extrusion.orientation.x	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC REGULAR VALUE = ["-"] DEC NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the

Attribute Name	varTo.stringValue
	TimeAnimationValueListEntry record.
extrusion.orientation.y	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC PURE FLOATING = ["-"] "." DEC NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
extrusion.orientation.z	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
extrusion.orientationangle	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
extrusion.rotationangle.x	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC_NUMBER = 1*DIGIT

Attribute Name	varTo.stringValue
	DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
extrusion.rotationangle.y	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC_EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
extrusion.rotationcenter.x	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL NUMBER = DEC REGULAR VALUE / DEC PURE FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC_NUMBER] [DEC_EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
extrusion.rotationcenter.y	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA OR NUMBER = SETFORMULA / REAL NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."] [DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.
extrusion.rotationcenter.z	MUST be a valid FORMULA_OR_NUMBER as specified in the following ABNF (specified in [RFC5234]) grammar: FORMULA_OR_NUMBER = SETFORMULA / REAL_NUMBER SETFORMULA = "(" FORMULA ")" REAL_NUMBER = DEC_REGULAR_VALUE / DEC_PURE_FLOATING DEC_REGULAR_VALUE = ["-"] DEC_NUMBER ["."]

Attribute Name	varTo.stringValue
	[DEC NUMBER] [DEC EXPONENT] DEC_PURE_FLOATING = ["-"] "." DEC_NUMBER [DEC EXPONENT] DEC_NUMBER = 1*DIGIT DEC_EXPONENT = ["-"] ("e" / "E") DEC_NUMBER The FORMULA rule is specified by the **varFormula** field of the TimeAnimationValueListEntry record.

When using attribute names in the following table, the **setBehaviorAtom.valueType** field MUST be TL_TABVT_Color.

Attribute Name	varTo.stringValue
ppt_c	MUST be a valid SETCOLOR as specified in the following ABNF (specified in [RFC5234]) grammar: SETCOLOR = "#" RED GREEN BLUE RED = 2HEXDIG GREEN = 2HEXDIG BLUE = 2HEXDIG
fillcolor	MUST be a valid SETCOLOR as specified in the following ABNF (specified in [RFC5234]) grammar: SETCOLOR = "#" RED GREEN BLUE RED = 2HEXDIG GREEN = 2HEXDIG BLUE = 2HEXDIG
style.color	MUST be a valid SETCOLOR as specified in the following ABNF (specified in [RFC5234]) grammar: SETCOLOR = "#" RED GREEN BLUE RED = 2HEXDIG GREEN = 2HEXDIG BLUE = 2HEXDIG
imageData.chromakey	MUST be a valid SETCOLOR as specified in the following ABNF (specified in [RFC5234]) grammar: SETCOLOR = "#" RED GREEN BLUE RED = 2HEXDIG GREEN = 2HEXDIG BLUE = 2HEXDIG
fill.color	MUST be a valid SETCOLOR as specified in the following ABNF (specified in [RFC5234]) grammar: SETCOLOR = "#" RED GREEN BLUE RED = 2HEXDIG

Attribute Name	varTo.stringValue
	GREEN = 2HEXDIG BLUE = 2HEXDIG
fill.color2	MUST be a valid SETCOLOR as specified in the following ABNF (specified in [RFC5234]) grammar: SETCOLOR = "#" RED GREEN BLUE RED = 2HEXDIG GREEN = 2HEXDIG BLUE = 2HEXDIG
stroke.color	MUST be a valid SETCOLOR as specified in the following ABNF (specified in [RFC5234]) grammar: SETCOLOR = "#" RED GREEN BLUE RED = 2HEXDIG GREEN = 2HEXDIG BLUE = 2HEXDIG
stroke.color2	MUST be a valid SETCOLOR as specified in the following ABNF (specified in [RFC5234]) grammar: SETCOLOR = "#" RED GREEN BLUE RED = 2HEXDIG GREEN = 2HEXDIG BLUE = 2HEXDIG
shadow.color	MUST be a valid SETCOLOR as specified in the following ABNF (specified in [RFC5234]) grammar: SETCOLOR = "#" RED GREEN BLUE RED = 2HEXDIG GREEN = 2HEXDIG BLUE = 2HEXDIG
shadow.color2	MUST be a valid SETCOLOR as specified in the following ABNF (specified in [RFC5234]) grammar: SETCOLOR = "#" RED GREEN BLUE RED = 2HEXDIG GREEN = 2HEXDIG BLUE = 2HEXDIG
extrusion.color	MUST be a valid SETCOLOR as specified in the following ABNF (specified in [RFC5234]) grammar: SETCOLOR = "#" RED GREEN BLUE RED = 2HEXDIG GREEN = 2HEXDIG

Attribute Name	varTo.stringValue
	BLUE = 2HEXDIG

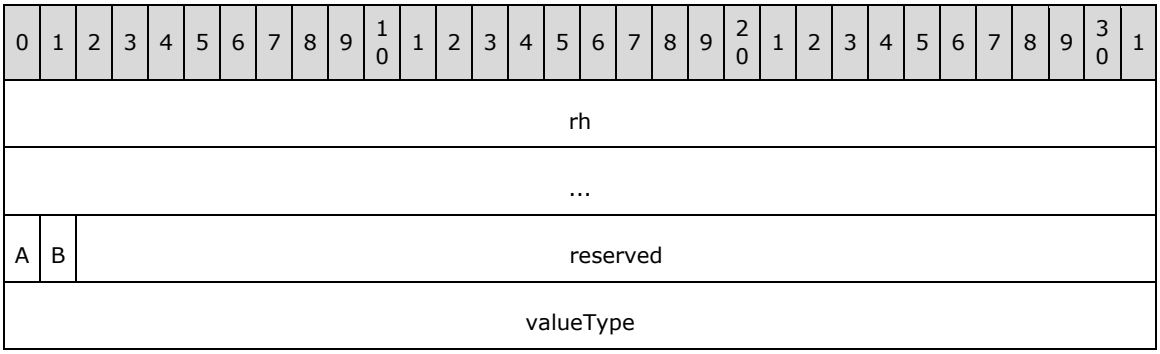
behavior (variable): A **TimeBehaviorContainer** record (section 2.8.34) that specifies the common behavior information.

2.8.70 TimeSetBehaviorAtom

Referenced by: TimeSetBehaviorContainer

An atom record that specifies animation information for an object or object property.

TimeAnimateBehaviorValueTypeEnum enumeration specifies the object or object property that will be animated.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeSetBehavior.
rh.recLen	MUST be 0x00000008.

A - fToPropertyUsed (1 bit): A bit that specifies whether the **varTo** of the **TimeSetBehaviorContainer** record (section 2.8.69) that contains this **TimeSetBehaviorAtom** is valid.

B - fValueTypePropertyUsed (1 bit): A bit that specifies whether **valueType** was explicitly set by a user interface action.

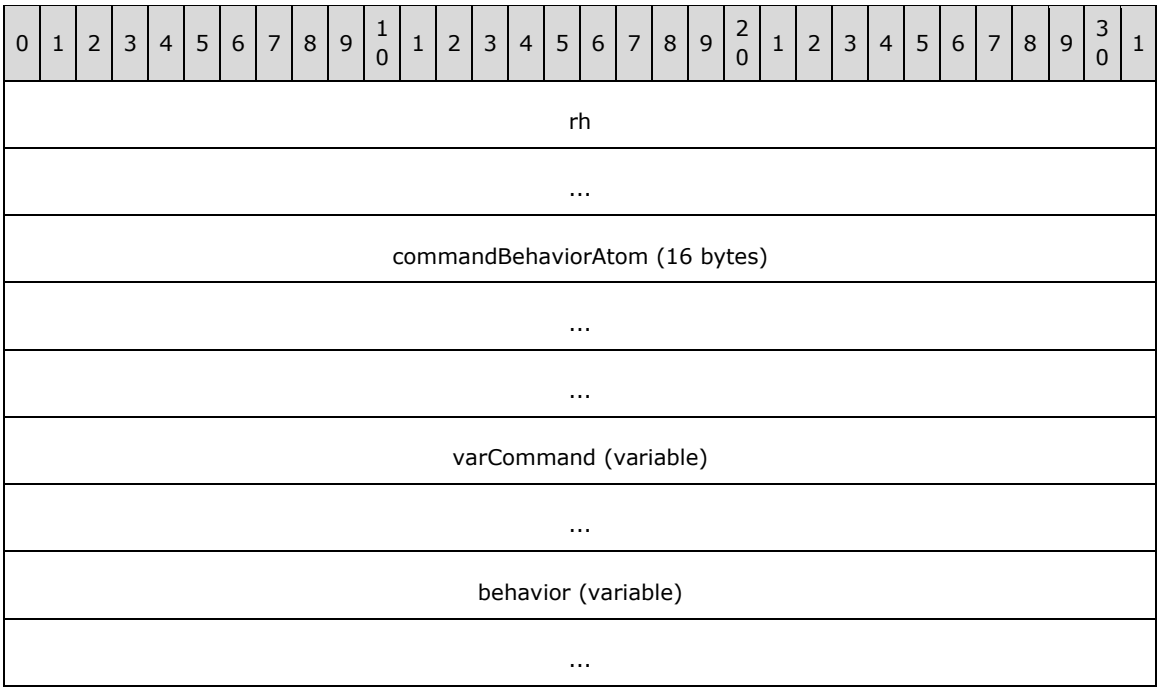
reserved (30 bits): MUST be zero, and MUST be ignored.

valueType (4 bytes): A **TimeAnimateBehaviorValueTypeEnum** enumeration that specifies the type of object or object property to which to apply an animation. It MUST be ignored if **fValueTypePropertyUsed** is **FALSE** and a value of TL_TABVT_Number MUST be used instead.

2.8.71 TimeCommandBehaviorContainer

Referenced by: ExtTimeNodeContainer, SubEffectContainer

A container record that specifies a command behavior that performs a command as an animation. There is no property to be animated in this behavior. The **behavior.stringList** field is ignored.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Subfields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeCommandBehaviorContainer.

commandBehaviorAtom (16 bytes): A TimeCommandBehaviorAtom record that specifies the property usage flag of the command behavior and the type of the command behavior.

varCommand (variable): An optional TimeVariantString record that specifies the command to be performed. It MUST be ignored if **commandBehaviorAtom.fCommandPropertyUsed** is FALSE. The **varCommand.rh.recInstance** sub-field MUST be 0x001.

When **commandBehaviorAtom.commandBehaviorType** is TL_TCBT_Event, the command MUST be "onstopaudio" that specifies to stop playing of all audio.

When **commandBehaviorAtom.commandBehaviorType** is TL_TCBT_Call, the command MUST be one from the following table.

Command	Meaning
play	Play corresponding media.
playFrom(s)	Play corresponding media starting from s, where s is the number of seconds from the beginning of the clip.

Command	Meaning
pause	Pause corresponding media.
resume	Resume playing of corresponding media.
stop	Stop playing of corresponding media.
togglePause	Play corresponding media if media is paused, or pause corresponding media if media is playing. If the corresponding media is not active, this command will restart the media and play from its beginning.

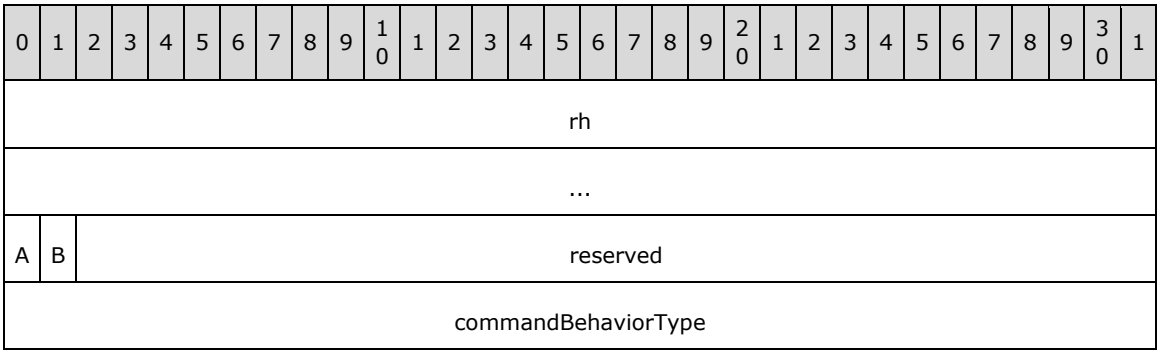
When **commandBehaviorAtom.commandBehaviorType** is TL_TCBT_OleVerb, the command MUST be the string representation of an integer that specifies the embedded object verb number that determines the action.

behavior (variable): A **TimeBehaviorContainer** record (section 2.8.34) that specifies the common behavior information.

2.8.72 TimeCommandBehaviorAtom

Referenced by: TimeCommandBehaviorContainer

An atom record that specifies the information of a command that is performed as an animation.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeCommandBehavior.
rh.recLen	MUST be 0x00000008.

A - fTypePropertyUsed (1 bit): A bit that specifies whether **commandBehaviorType** was explicitly set by a user interface action.

B - fCommandPropertyUsed (1 bit): A bit that specifies whether **varCommand** of the **TimeCommandBehaviorContainer** record (section 2.8.71) that contains this **TimeCommandBehaviorAtom** is valid.

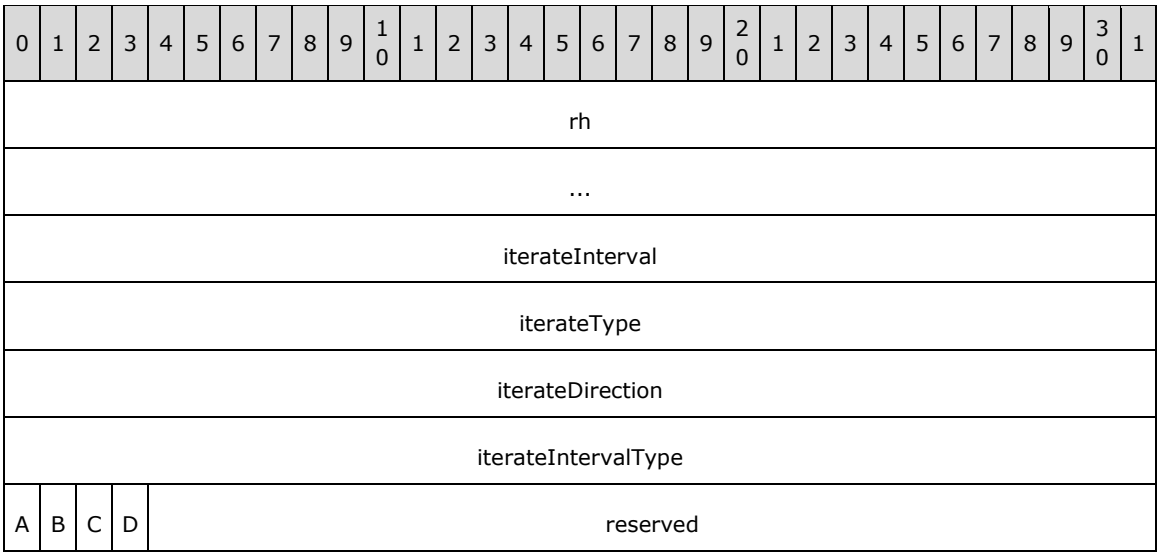
reserved (30 bits): MUST be zero, and MUST be ignored.

commandBehaviorType (4 bytes): A **TimeCommandBehaviorTypeEnum** enumeration that specifies the type of the command. It MUST be ignored if **fTypePropertyUsed** is **FALSE** and a value of TL_TCBT_Call MUST be used instead.

2.8.73 TimeIterateDataAtom

Referenced by: ExtTimeNodeContainer

An atom record that specifies how an animation is applied to sub-elements of the target object for a repeated effect. It can be applied to the letters, words, or shapes within a target object.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeIterateData.
rh.recLen	MUST be 0x00000014.

iterateInterval (4 bytes): An unsigned integer that specifies the interval time of iterations, which can be either absolute time or a percentage as specified in **iterateIntervalType**. It MUST be ignored if **fIterateIntervalPropertyUsed** is **FALSE** and a value of 0x00000000 MUST be used instead.

iterateType (4 bytes): An unsigned integer that specifies the type of iteration behavior. It MUST be ignored if **fIterateTypePropertyUsed** is **FALSE** and a value of 0x00000000 MUST be used instead. It MUST be a value from the following table.

Value	Meaning
0x00000000	All at once: all sub-elements animate together with no interval time.
0x00000001	By word: sub-elements are words.
0x00000002	By letter: sub-elements are letters.

iterateDirection (4 bytes): An unsigned integer that specifies the direction of the iteration behavior. It MUST be ignored if **fIterateDirectionPropertyUsed** is **FALSE** and a value of 0x00000001 MUST be used instead. It MUST be a value from the following table.

Value	Meaning
0x00000000	Backwards: from the last sub-element to the first sub-element.
0x00000001	Forwards: from the first sub-element to the last sub-element.

iterateIntervalType (4 bytes): An unsigned integer that specifies the type of interval time as specified in **iterateInterval**. It MUST be ignored if **fIterateIntervalTypePropertyUsed** is **FALSE** and a value of 0x00000000 MUST be used instead. It MUST be a value from the following table.

Value	Meaning
0x00000000	Seconds: **iterateInterval** is absolute time in milliseconds.
0x00000001	Percentage: **iterateInterval** is a percentage of animation duration, in tenths of a percent.

A - fIterateDirectionPropertyUsed (1 bit): A **bit** that specifies whether **iterateDirection** was explicitly set by a user interface action.

B - fIterateTypePropertyUsed (1 bit): A bit that specifies whether **iterateType** was explicitly set by a user interface action.

C - fIterateIntervalPropertyUsed (1 bit): A bit that specifies whether **iterateInterval** was explicitly set by a user interface action.

D - fIterateIntervalTypePropertyUsed (1 bit): A bit that specifies whether **iterateIntervalType** was explicitly set by a user interface action.

reserved (28 bits): MUST be zero, and MUST be ignored.

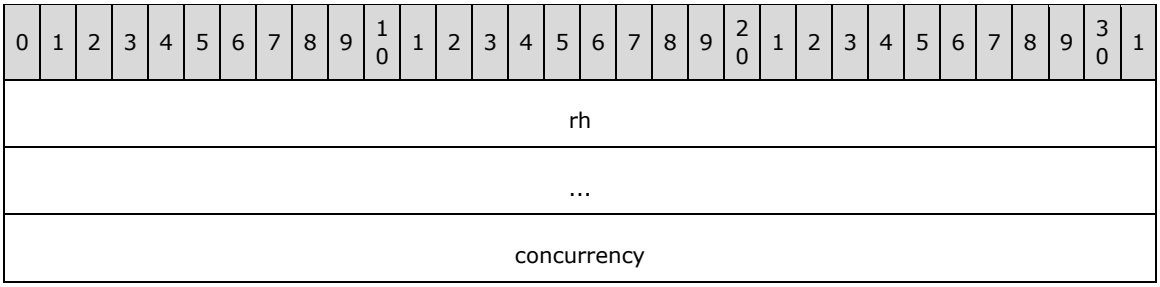
2.8.74 TimeSequenceDataAtom

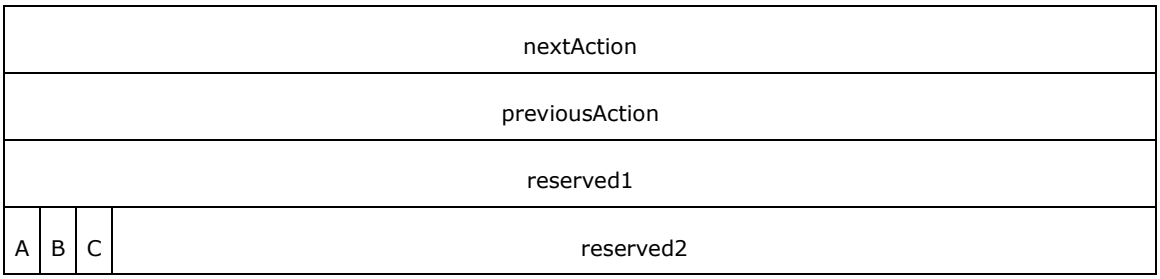
Referenced by: *ExtTimeNodeContainer*

An atom record that specifies sequencing information for the child nodes of a time node. Each child can only be activated after its prior sibling has been activated.

Let the *corresponding time node* be specified by the **ExtTimeNodeContainer** record (section 2.8.15) that contains this **TimeSequenceDataAtom** record.

Let the *corresponding next time condition list* be specified by the **rgBeginTimeCondition** field in *corresponding time node*, which specifies the time conditions to activate the next child time node.





rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeSequenceData.
rh.recLen	MUST be 0x00000014.

concurrency (4 bytes): An **unsigned integer** that specifies the concurrency behavior of the child nodes of the *corresponding time node*. It MUST be ignored if **fConcurrencyPropertyUsed** is **FALSE** and a value of 0x00000000 MUST be used instead. It MUST be a value from the following table.

Value	Meaning
0x00000000	No concurrency: the next child is activated only after the current child ends and conditions in the *corresponding next time condition list* are met.
0x00000001	Concurrency enabled: the next child can be activated after the current child is activated and conditions in the *corresponding next time condition list* are met.

nextAction (4 bytes): An **unsigned integer** that specifies actions when traversing forward in the sequence of child nodes of the *corresponding time node*. It MUST be ignored if **fNextActionPropertyUsed** is **FALSE** and a value of 0x00000000 MUST be used instead. It MUST be a value from the following table.

Value	Meaning
0x00000000	Take no action.
0x00000001	Traverse forward the current child node along the timeline to a natural end time. The natural end time of a child node is the time when the child node will end without interventions. If the end time is infinite, the child node will never stop. The natural end time of the child node is specified as the latest non-infinite end time of its child nodes.

previousAction (4 bytes): An **unsigned integer** that specifies actions when traversing backward in the sequence of child nodes of the *corresponding time node*. It MUST be ignored if

fPreviousActionPropertyUsed is **FALSE** and a value of 0x0000000 MUST be used instead. It MUST be a value from the following table.

Value	Meaning
0x00000000	Take no action.
0x00000001	Continue backwards in the sequence until reaching a child that starts only on the next time condition as specified in the *corresponding next time condition list*.

reserved1 (4 bytes): MUST be zero, and MUST be ignored.

A - fConcurrencyPropertyUsed (1 bit): A bit that specifies whether **concurrency** was explicitly set by a user interface action.

B - fNextActionPropertyUsed (1 bit): A bit that specifies whether **nextAction** was explicitly set by a user interface action.

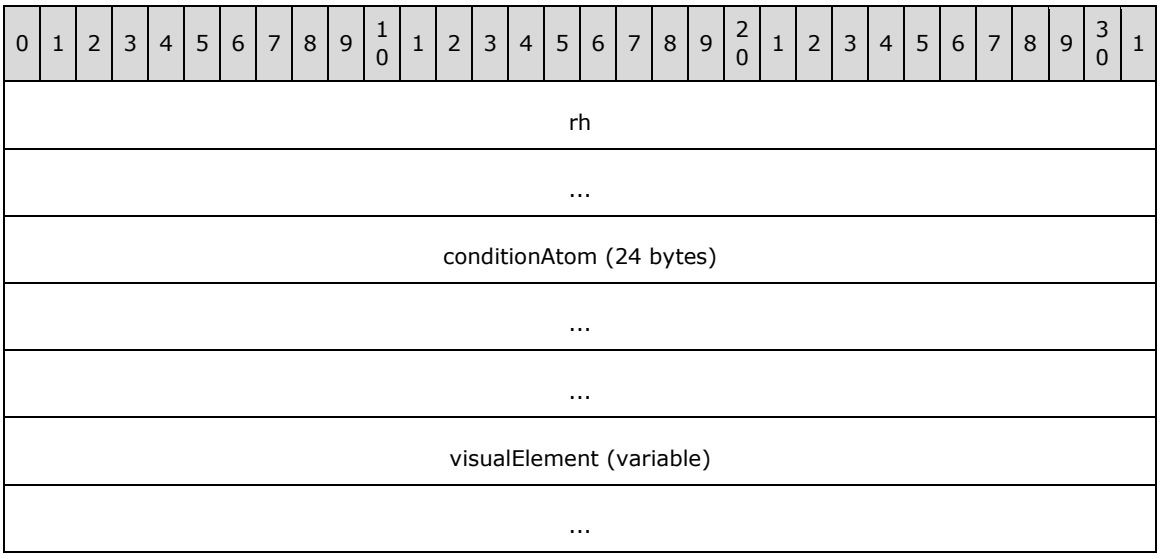
C - fPreviousActionPropertyUsed (1 bit): A **bit** that specifies whether **previousAction** was explicitly set by a user interface action.

reserved2 (29 bits): MUST be zero, and MUST be ignored.

2.8.75 TimeConditionContainer

Referenced by: ExtTimeNodeContainer, SubEffectContainer

A container record that specifies a time condition of a time node.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Subfields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be greater than or equal to 0x000 and less than or equal to 0x005.
rh.recType	MUST be an RT_TimeConditionContainer.

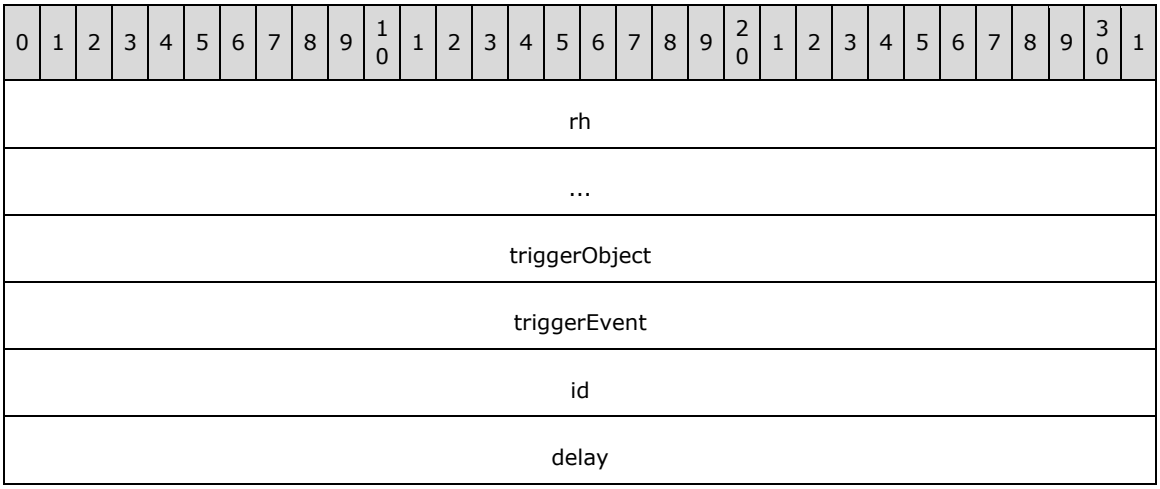
conditionAtom (24 bytes): A TimeConditionAtom record that specifies the information that is used to evaluate the time condition.

visualElement (variable): An optional **ClientVisualElementContainer** record (section 2.8.44) that specifies the target object that participates in the evaluation of the time condition. It MUST exist if and only if **conditionAtom.triggerObject** is TL_TOT_VisualElement.

2.8.76 TimeConditionAtom

Referenced by: TimeConditionContainer

An atom record that specifies the information used to evaluate when a condition will be true.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TimeCondition.
rh.recLen	MUST be 0x00000010.

triggerObject (4 bytes): A **TriggerObjectEnum** enumeration that specifies the type of target that participates in the evaluation of the condition.

triggerEvent (4 bytes): An unsigned integer that specifies the event that causes the condition to be **TRUE**. It MUST be a value from the following table.

Value	Meaning
0x00000000	None.
0x00000001	OnBegin event that occurs on the specified target.
0x00000003	Start of the time node that is specified by **id.**
0x00000004	End of the time node that is specified by **id.**
0x00000005	Mouse click.
0x00000007	Mouse over.

Value	Meaning
0x00000009	OnNext event that occurs on the specified target.
0x0000000A	OnPrev event that occurs on the specified target.
0x0000000B	Stop audio event that occurs when an "onstopaudio" command is fired.

id (4 bytes): An unsigned integer that specifies the target that participates in the evaluation of the condition.

When **triggerObject** is TL_TOT_TimeNode, this field specifies the time node identifier.

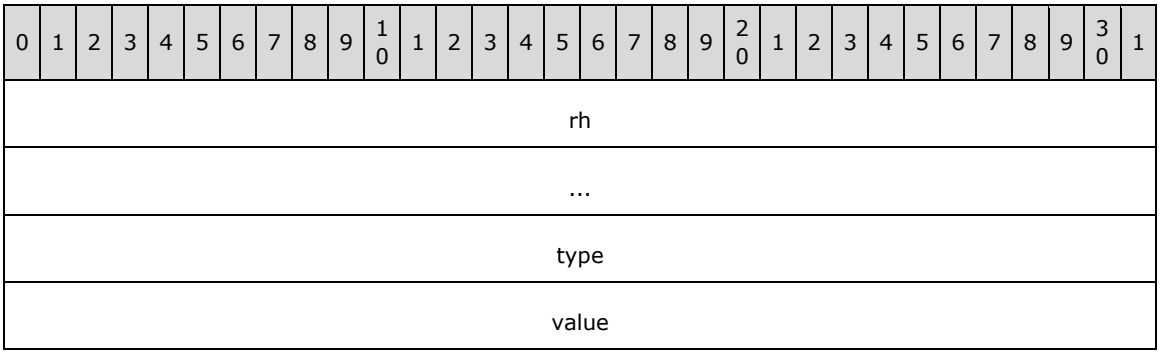
When **triggerObject** is TL_TOT_RuntimeNodeRef, this field MUST be 0x00000002 that specifies that all child time node of the **ExtTimeNodeContainer** record (section 2.8.15) or **SubEffectContainer** record (section 2.8.16) that contains this record are the target.

delay (4 bytes): A signed integer that specifies the offset time, in milliseconds, that sets when the condition will become **TRUE**.

2.8.77 TimeModifierAtom

Referenced by: ExtTimeNodeContainer, SubEffectContainer

An atom record that specifies the type of data to be modified and the new data value.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be RT_TimeModifier.
rh.recLen	MUST be 0x00000008.

type (4 bytes): An **unsigned integer** that specifies the type of data to be modified. It MUST be a value from the following table.

Value	Meaning
0x00000000	Repeat count.
0x00000001	Repeat duration.
0x00000002	Speed.
0x00000003	Accelerate.

Value	Meaning
0x00000004	Decelerate.
0x00000005	Automatic reverse.

value (4 bytes): An **unsigned integer** that specifies the new value.

2.8.78 TimeVariant

Referenced by: TimeAnimationValueListEntry

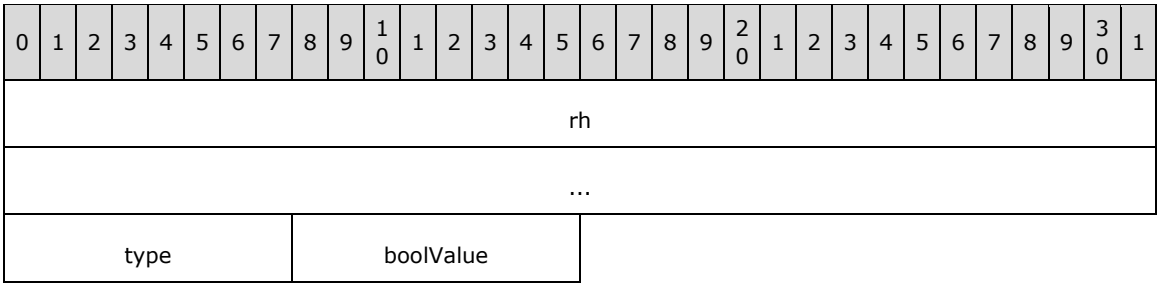
A variable type record whose type and meaning are dictated by the value of the **type** field of any of these four structures, as specified by the following table.

Value	Meaning
TL_TVT_Bool	A TimeVariantBool record that specifies a Boolean value.
TL_TVT_Int	A TimeVariantInt that specifies an integer value.
TL_TVT_Float	A TimeVariantFloat record that specifies a floating-point number.
TL_TVT_String	A TimeVariantString record that specifies a string.

2.8.79 TimeVariantBool

Referenced by: TimeVariant, TimeVariant4Behavior, TimeVariant4TimeNode

An atom record that specifies a Boolean value.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be an RT_TimeVariant.
rh.recLen	MUST be 0x00000002.

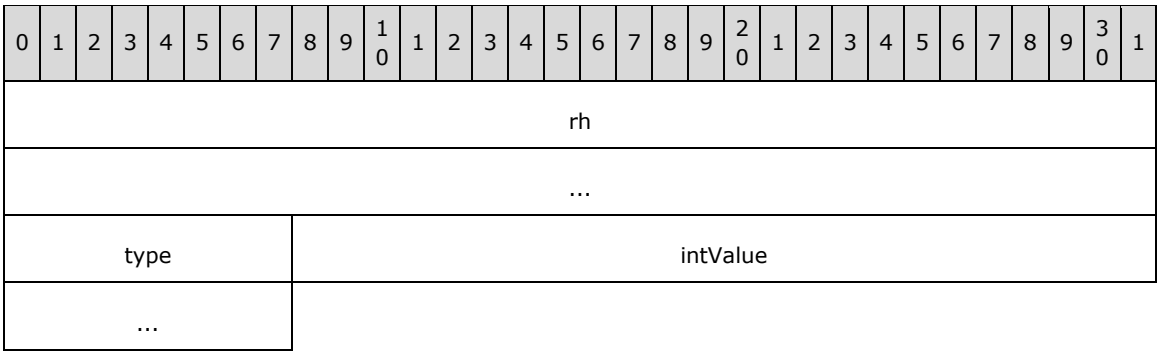
type (1 byte): A **TimeVariantTypeEnum** enumeration that specifies the data type of this record. It MUST be TL_TVT_Bool.

boolValue (1 byte): A **bool1** (section 2.2.2) that specifies the value.

2.8.80 TimeVariantInt

Referenced by: TimeMotionBehaviorContainer, TimeVariant, TimeVariant4TimeNode

An atom record that specifies a signed integer value.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be an RT_TimeVariant.
rh.recLen	MUST be 0x00000005.

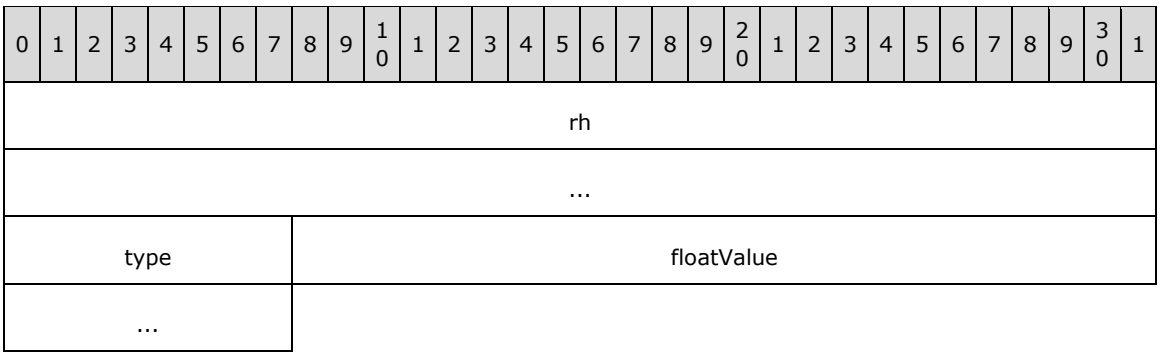
type (1 byte): A **TimeVariantTypeEnum** enumeration that specifies the data type of this record. It MUST be TL_TVT_Int.

intValue (4 bytes): A signed integer that specifies the value.

2.8.81 TimeVariantFloat

Referenced by: TimeEffectBehaviorContainer, TimeVariant, TimeVariant4Behavior, TimeVariant4TimeNode

An atom record that specifies a floating-point number.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be an RT_TimeVariant.
rh.recLen	MUST be 0x00000005.

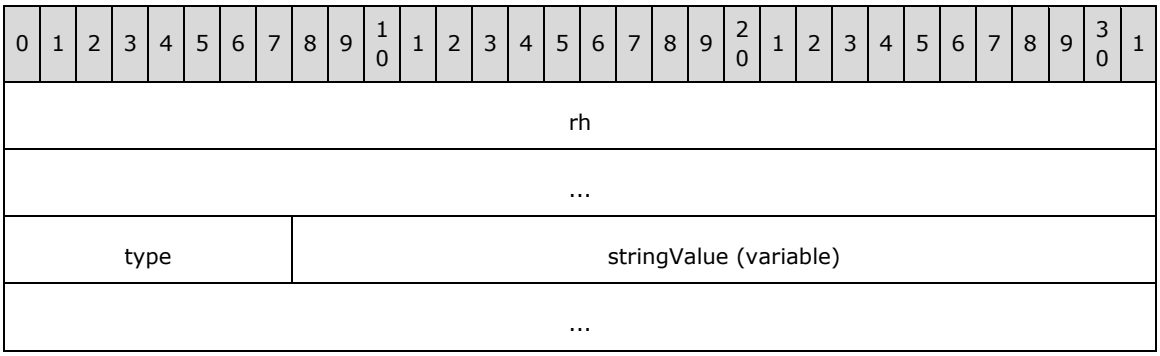
type (1 byte): A **TimeVariantTypeEnum** enumeration that specifies the data type of this record. It MUST be TL_TVT_Float.

floatValue (4 bytes): A floating-point number that specifies the value.

2.8.82 TimeVariantString

Referenced by: TimeAnimateBehaviorContainer, TimeAnimationValueListEntry, TimeCommandBehaviorContainer, TimeEffectBehaviorContainer, TimeMotionBehaviorContainer, TimeSetBehaviorContainer, TimeStringListContainer, TimeVariant, TimeVariant4Behavior

An atom record that specifies a Unicode string.



rh (8 bytes): A RecordHeader structure that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be an RT_TimeVariant.
rh.recLen	MUST be an odd number.

type (1 byte): A **TimeVariantTypeEnum** enumeration that specifies the data type of this record. It MUST be TL_TVT_String.

stringValue (variable): A UnicodeString that specifies the value. The length, in bytes, of the field is specified by the following formula:

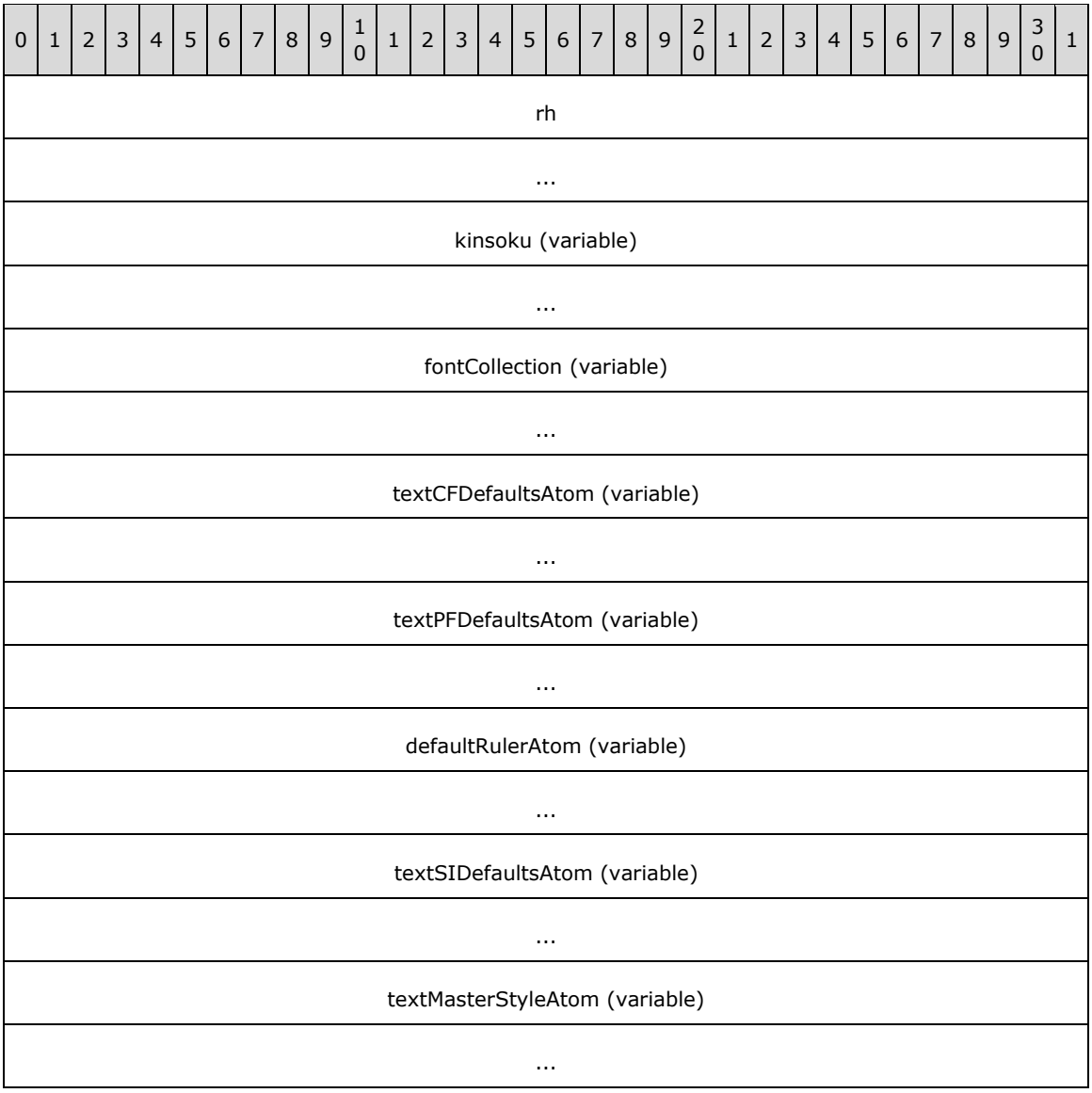
```
rh.recLen - 1.
```

2.9 Text Types

2.9.1 DocumentTextInfoContainer

Referenced by: DocumentContainer

A container record that specifies default settings and user preferences for text.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_Environment.

kinsoku (variable): An optional **KinsokuContainer** record (section 2.9.2) that specifies the user preferences for East Asian text line breaking settings.

fontCollection (variable): An optional **FontCollectionContainer** record (section 2.9.8) that specifies the collection of fonts used.

textCFDefaultsAtom (variable): An optional TextCFExceptionAtom record that specifies default settings for character-level style and formatting.

textPFDefaultsAtom (variable): An optional TextPFExceptionAtom record that specifies default settings for paragraph-level style and formatting.

defaultRulerAtom (variable): An optional DefaultRulerAtom record that specifies the default ruler and corresponding options.

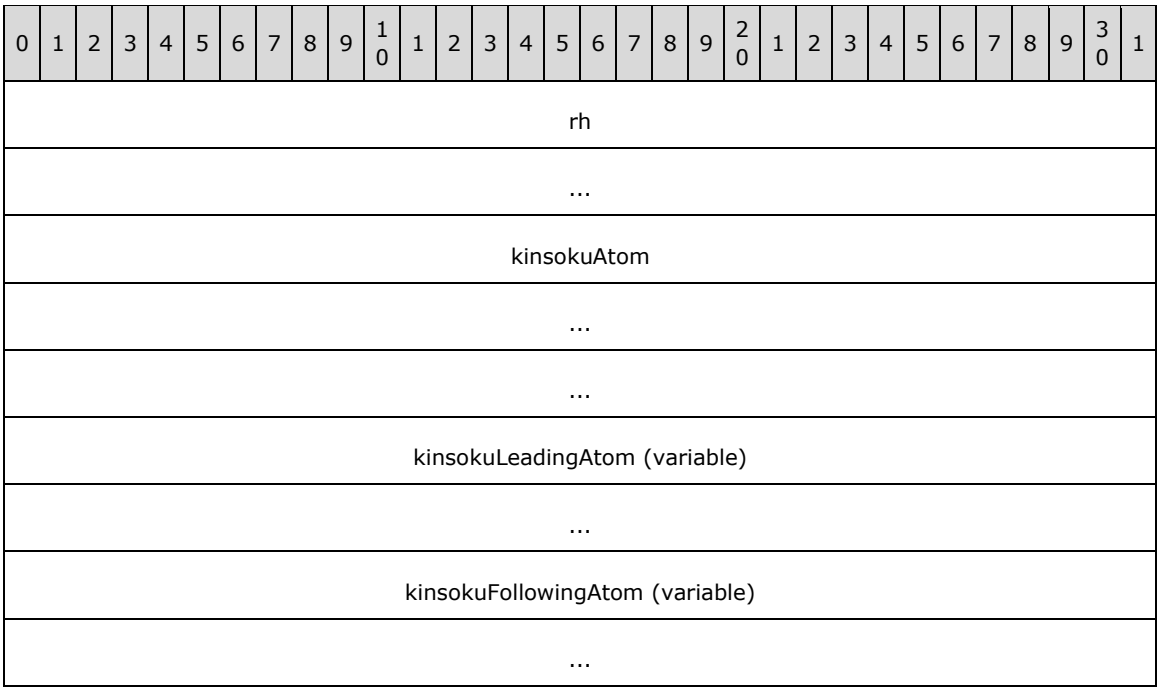
textSIDefaultsAtom (variable): A TextSIExceptionAtom record that specifies default settings for language and spelling information.

textMasterStyleAtom (variable): A TextMasterStyleAtom record that specifies the character-level and paragraph-level formatting of a main master slide.

2.9.2 KinsokuContainer

Referenced by: DocumentTextInfoContainer

A container record that specifies user preferences for East Asian text line breaking settings.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x002.
rh.recType	MUST be an RT_Kinsoku.

kinsokuAtom (12 bytes): A KinsokuAtom record that specifies the type of East Asian text line breaking settings.

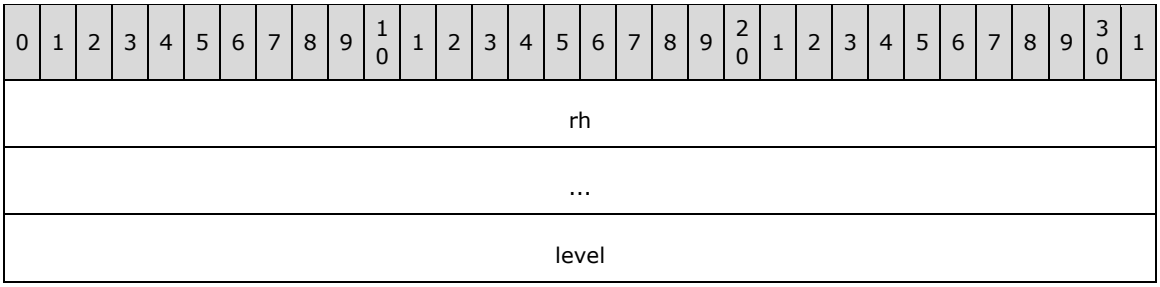
kinsokuLeadingAtom (variable): An optional KinsokuLeadingAtom record that specifies a list of characters that cannot appear immediately after a line break of East Asian text. It MUST exist if and only if **kinsokuAtom.level** is 0x00000002. If this field exists and the document contains a **Kinsoku9Container** record (section 2.9.6), the **Kinsoku9Container** record MUST NOT contain a **kinsokuLeadingAtom** field.

kinsokuFollowingAtom (variable): An optional KinsokuFollowingAtom record that specifies a list of characters that cannot appear immediately before a line break of East Asian text. It MUST exist if and only if **kinsokuAtom.level** is 0x00000002. If this field exists and the document contains a **Kinsoku9Container** record, the **Kinsoku9Container** record MUST NOT contain a **kinsokuFollowingAtom** field.

2.9.3 KinsokuAtom

Referenced by: KinsokuContainer

An atom record that specifies the type of East Asian text line breaking.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x003.
rh.recType	MUST be an RT_KinsokuAtom.
rh.recLen	MUST be 0x00000004.

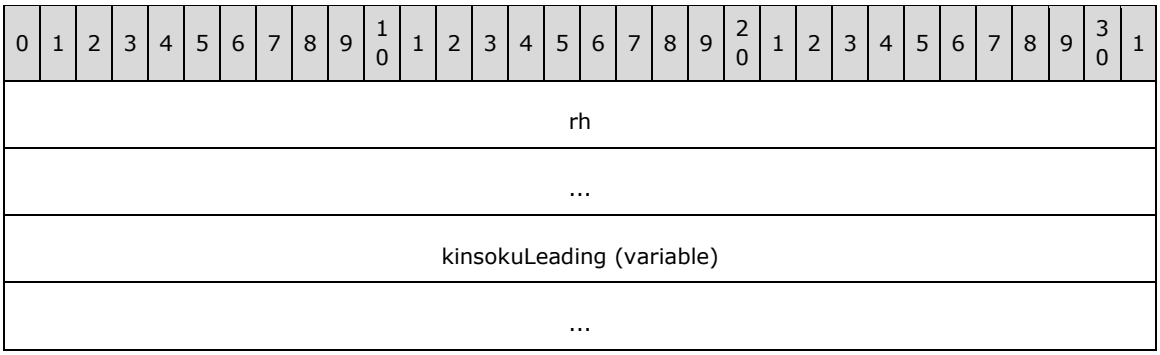
level (4 bytes): A signed integer that specifies the type of East Asian text line breaking. It MUST be a value from the following table.

Value	Meaning
0x00000000	Use standard line breaking settings.
0x00000001	Use strict line breaking settings for Japanese text.
0x00000002	Use the customized line breaking settings specified by the **kinsokuLeadingAtom** and **kinsokuFollowingAtom** fields of the **KinsokuContainer** (section 2.9.2) that contains this KinsokuAtom.

2.9.4 KinsokuLeadingAtom

Referenced by: Kinsoku9Container, KinsokuContainer

An atom record that specifies a list of characters that cannot appear immediately after a line break of East Asian text.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

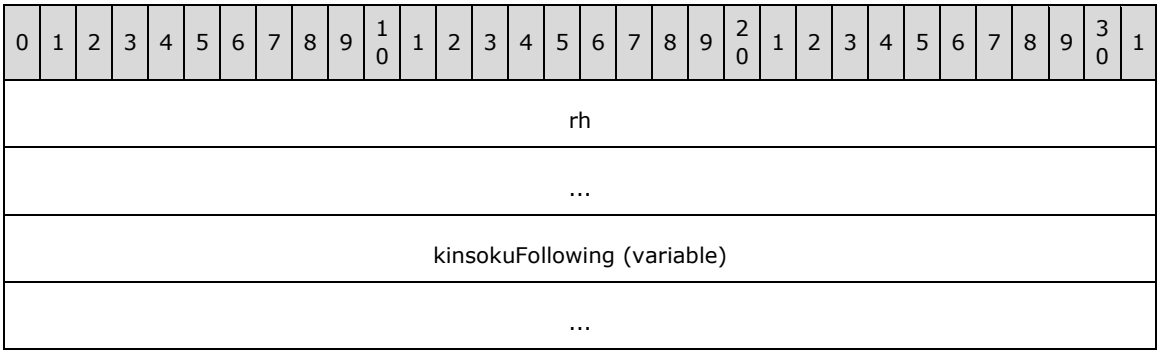
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number

kinsokuLeading (variable): An array of **UTF-16 Unicode** [RFC2781] characters that specifies the list of characters.

2.9.5 KinsokuFollowingAtom

Referenced by: Kinsoku9Container, KinsokuContainer

An atom record that specifies a list of characters that cannot appear immediately before a line break of East Asian text.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

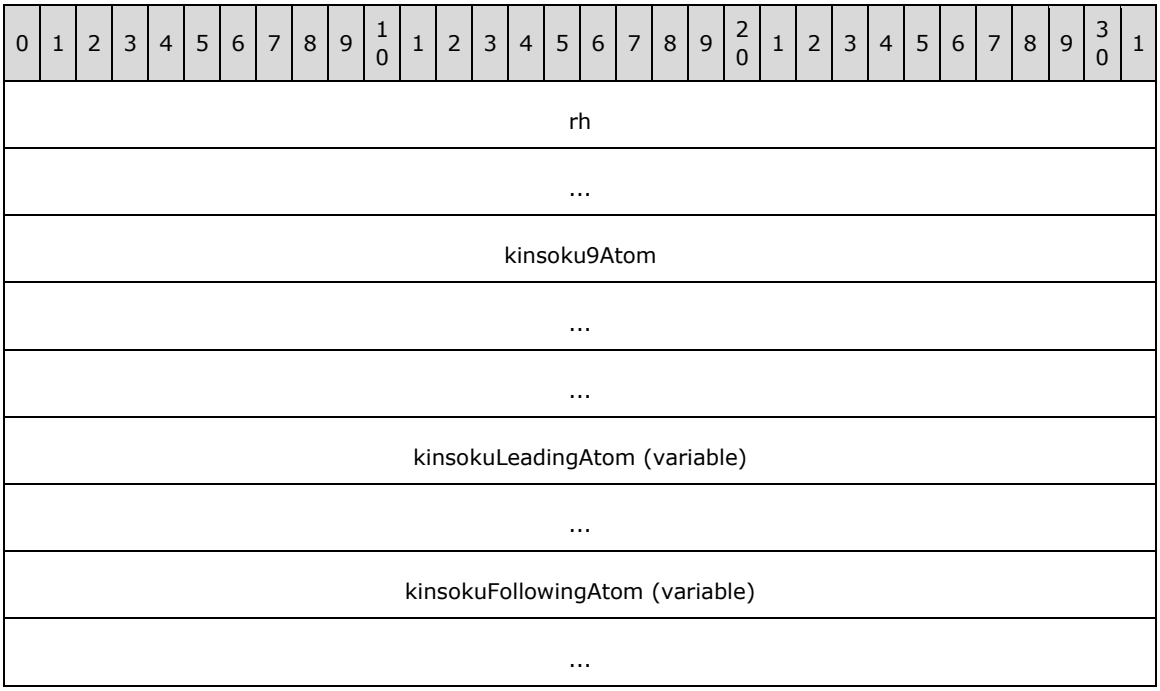
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be an **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number

kinsokuFollowing (variable): An array of **UTF-16 Unicode** [RFC2781] characters that specifies the list of characters.

2.9.6 Kinsoku9Container

Referenced by: PP9DocBinaryTagExtension

A container record that specifies the user preferences for East Asian text line break settings for the following languages: Japanese, Korean, Simplified Chinese and Traditional Chinese.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x002.
rh.recType	MUST be an RT_Kinsoku.

kinsoku9Atom (12 bytes): A Kinsoku2.9.7Atom record that specifies the types of East Asian text line breaking for the following languages: Japanese, Korean, Simplified Chinese and Traditional Chinese. This field takes precedence over the settings specified by the **kinsokuAtom** field of the **KinsokuContainer** record (section 2.9.2) in the document. At most one of the following sub-fields MUST be set to 0x00000002: **kinsoku9Atom.korLevel**, **kinsoku9Atom.scLevel**, **kinsoku9Atom.tcLevel**, or **kinsoku9Atom.jpnLevel**.

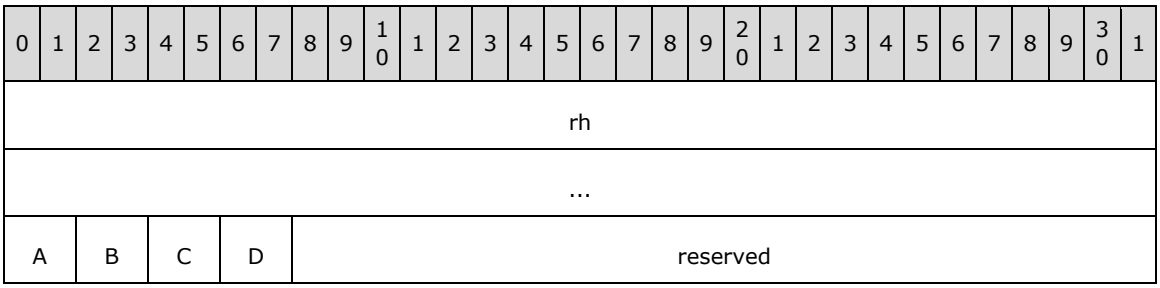
kinsokuLeadingAtom (variable): An optional KinsokuLeadingAtom record that specifies a list of characters that cannot appear immediately after a line break of East Asian text. It MUST exist if and only if the **kinsokuLeadingAtom** field of the **KinsokuContainer** record does not exist and one of **kinsoku9Atom.korLevel, kinsoku9Atom.scLevel, kinsoku9Atom.tcLevel**, or **kinsoku9Atom.jpnLevel** is 0x00000002.

kinsokuFollowingAtom (variable): An optional KinsokuFollowingAtom record that specifies a list of characters that cannot appear immediately before a line break of East Asian text. It MUST exist if and only if the **kinsokuFollowingAtom** field of the **KinsokuContainer** record does not exist and one of **kinsoku9Atom.korLevel, kinsoku9Atom.scLevel, kinsoku9Atom.tcLevel**, or **kinsoku9Atom.jpnLevel** is 0x00000002.

2.9.7 Kinsoku9Atom

Referenced by: Kinsoku9Container

An atom record that specifies information about the types of East Asian text line breaking for the following languages: Japanese, Korean, Simplified Chinese and Traditional Chinese.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x003.
rh.recType	MUST be an RT_KinsokuAtom.
rh.recLen	MUST be 0x00000004.

A - korLevel (2 bits): An **unsigned integer** that specifies the type of East Asian text line breaking applied to Korean text. It MUST be a value from the following table:

Value	Meaning
0x00000000	Use standard line breaking settings.
0x00000002	Use the customized line breaking settings specified by the KinsokuLeadingAtom and KinsokuFollowingAtom records contained in either the KinsokuContainer (section 2.9.2) or **Kinsoku9Container** record (section 2.9.6).

B - scLevel (2 bits): An **unsigned integer** that specifies the type of East Asian text line breaking applied to Simplified Chinese text. It MUST be a value from the following table.

Value	Meaning
0x00000000	Use standard line breaking settings.
0x00000002	Use the customized line breaking settings specified by the KinsokuLeadingAtom and KinsokuFollowingAtom records contained in either the **KinsokuContainer** or **Kinsoku9Container** record.

C - tcLevel (2 bits): An **unsigned integer** that specifies the type of East Asian text line breaking applied to Traditional Chinese text. It MUST be a value from the following table.

Value	Meaning
0x00000000	Use standard line breaking settings.
0x00000002	Use the customized line breaking settings specified by the KinsokuLeadingAtom and KinsokuFollowingAtom records contained in either the **KinsokuContainer** or **Kinsoku9Container** record.

D - jpnLevel (2 bits): An **unsigned integer** that specifies the type of East Asian text line breaking applied to Japanese text. It MUST be a value from the following table.

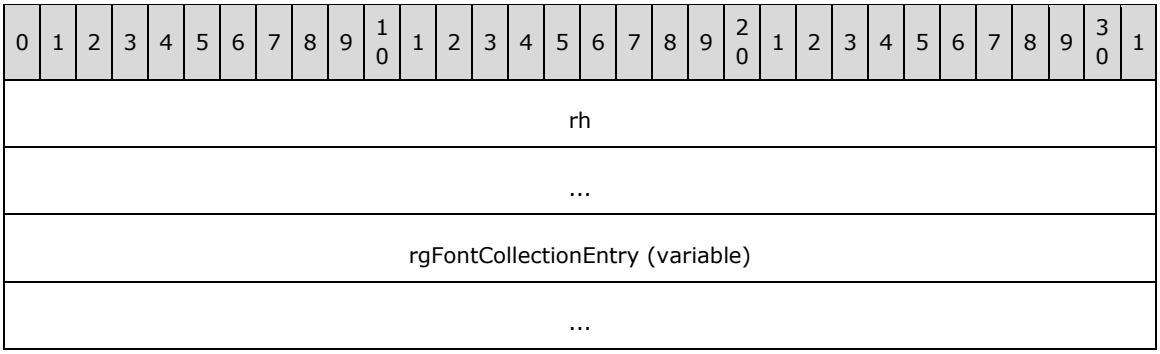
Value	Meaning
0x00000000	Use standard line breaking settings.
0x00000001	Use strict line breaking settings.
0x00000002	Use the customized line breaking settings specified by the KinsokuLeadingAtom and KinsokuFollowingAtom records contained in either the **KinsokuContainer** or **Kinsoku9Container** record.

reserved (24 bits): MUST be zero and MUST be ignored.

2.9.8 FontCollectionContainer

Referenced by: DocumentTextInfoContainer

A container record that specifies information about fonts in the presentation.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

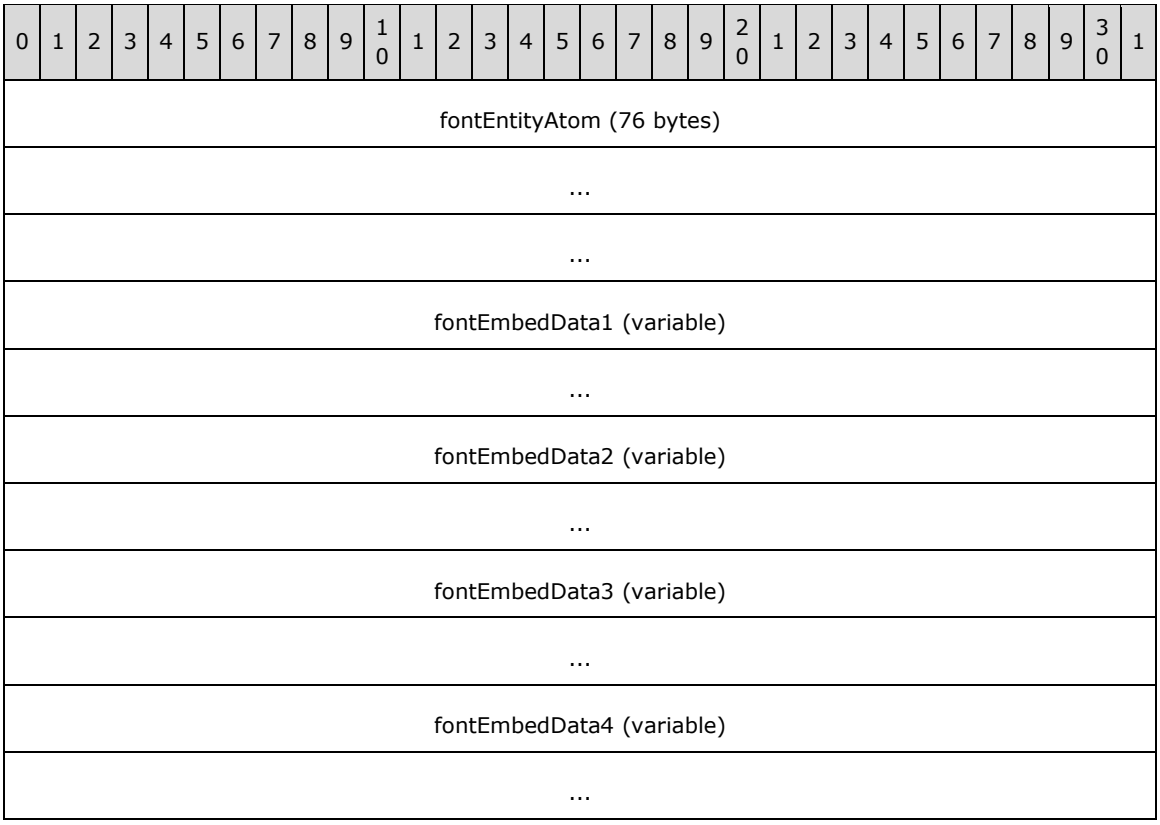
Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_FontCollection.

rgFontCollectionEntry (variable): An array of FontCollectionEntry structures that specifies information about the fonts. The size, in bytes, of the array is specified by **rh.recLen**.

2.9.9 FontCollectionEntry

Referenced by: FontCollection10Container, FontCollectionContainer

A structure that specifies information about a font contained in the presentation.



fontEntityAtom (76 bytes): A FontEntityAtom record that specifies the required attributes of the font.

fontEmbedData1 (variable): An optional FontEmbedDataBlob record that specifies the plain style data of an embedded font.

fontEmbedData2 (variable): An optional FontEmbedDataBlob record that specifies the bold style data of an embedded font.

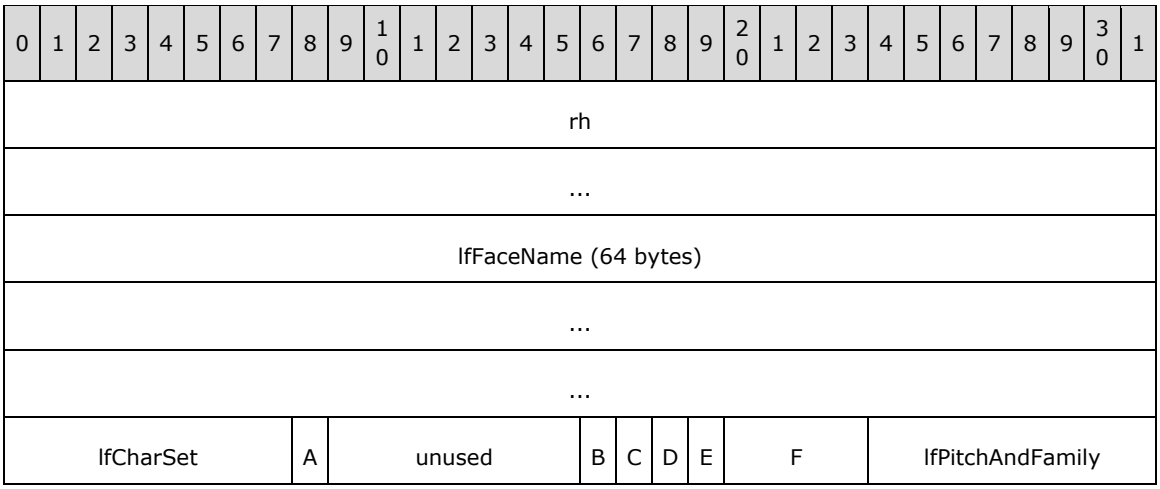
fontEmbedData3 (variable): An optional FontEmbedDataBlob record that specifies the italic style data of an embedded font.

fontEmbedData4 (variable): An optional FontEmbedDataBlob record that specifies the bold and italic style data of an embedded font.

2.9.10 FontEntityAtom

Referenced by: FontCollectionEntry

An atom record that specifies the information needed to define the attributes of a font, such as typeface name, character set, and so on, and corresponds in part to a Windows Logical Font (**LOGFONT**) structure [MC-LOGFONT].



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be greater than or equal to 0 and less than or equal to 128.
rh.recType	MUST be an RT_FontEntityAtom.
rh.recLen	MUST be equal to 0x00000044.

lfFaceName (64 bytes): A char2.2.4 that specifies the typeface name of the font. It corresponds to the **lfFaceName** field of the **LOGFONT** structure. The length of this string must not exceed 32 characters, including the terminating **null** character.

lfCharSet (1 byte): An unsigned byte that specifies the character set. It corresponds to the **lfCharSet** field of the **LOGFONT** structure.

A - fEmbedSubsetted (1 bit): A bit that specifies whether a subset of this font is embedded.

unused (7 bits): Undefined and MUST be ignored.

B - rasterFontType (1 bit): A bit that specifies whether the font is a raster font.

C - deviceFontType (1 bit): A bit that specifies whether the font is a device font.

D - truetypeFontType (1 bit): A bit that specifies whether the font is a TrueType font.

E - fNoFontSubstitution (1 bit): A bit that specifies whether font substitution logic is not applied for this font.

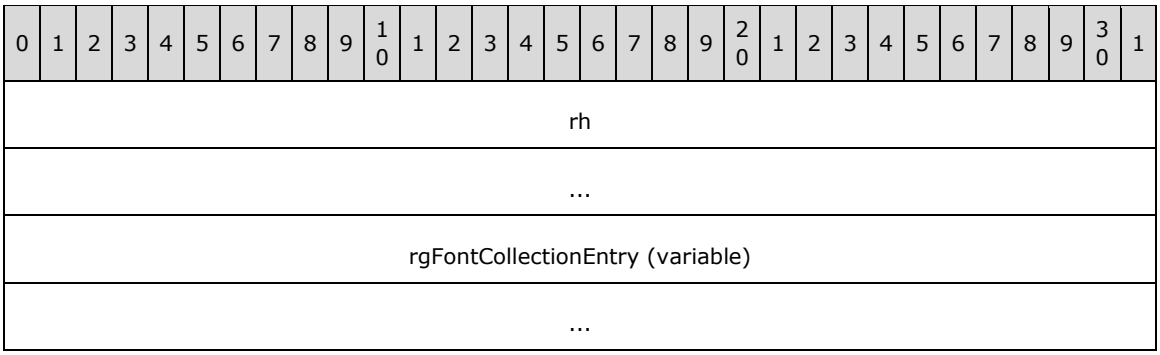
F - reserved (4 bits): MUST be zero and MUST be ignored.

lfPitchAndFamily (1 byte): An unsigned byte that specifies the pitch and family of the font. It corresponds to the **lfPitchAndFamily** field of the **LOGFONT** structure.

2.9.11 FontCollection10Container

Referenced by: PP10DocBinaryTagExtension

A container record that specifies information about fonts in the presentation for international support.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

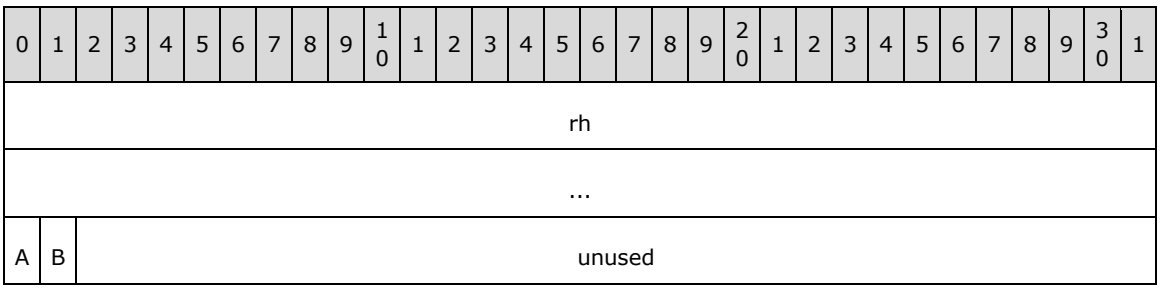
Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be RT_FontCollection10.

rgFontCollectionEntry (variable): An array of FontCollectionEntry structures that specifies information about the fonts. The size, in bytes, of the array is specified by **rh.recLen**.

2.9.12 FontEmbedFlags10Atom

Referenced by: PP10DocBinaryTagExtension

An atom record that specifies information about how font data is embedded.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_FontEmbedFlags10Atom.
rh.recLen	MUST be 0x00000004.

A - fSubset (1 bit): A bit that specifies whether embedded fonts contain data for only those characters in use.

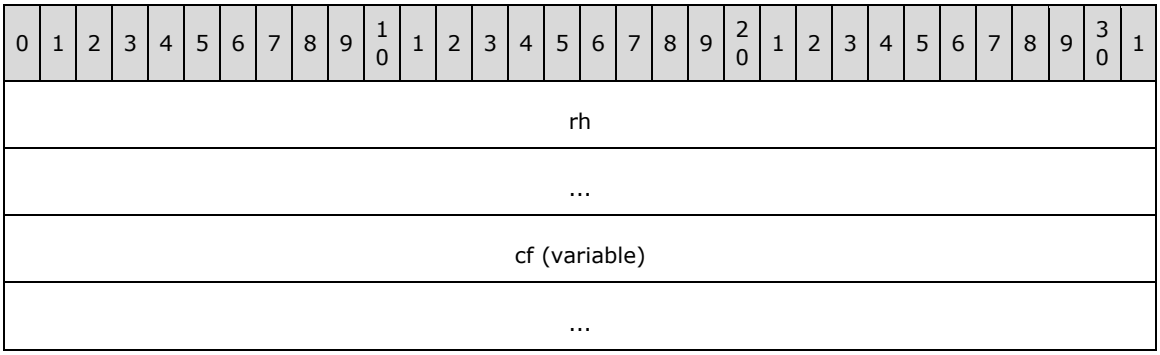
B - fSubsetOptionConfirmed (1 bit): A bit that specifies whether the user has confirmed the choice of **fSubset** in the user interface.

unused (30 bits): Undefined and MUST be ignored.

2.9.13 TextCFExceptionAtom

Referenced by: DocumentTextInfoContainer

An atom record that specifies character-level style and formatting.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

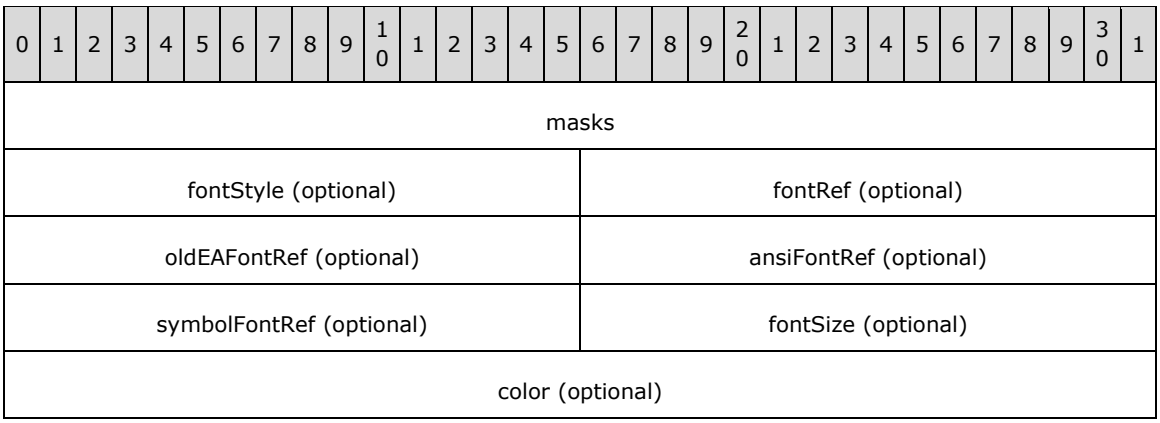
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TextCharFormatExceptionAtom.

cf (variable): A TextCFException structure that specifies character-level style and formatting.

2.9.14 TextCFException

Referenced by: TextCFExceptionAtom, TextCFRun, TextMasterStyleLevel

A structure that specifies character-level style and formatting, font information, coloring and positioning.



position (optional)

masks (4 bytes): A CFMasks structure that specifies whether certain fields in this **TextCFException** record exist and are valid. Sub-fields are further specified in the following table.

Field	Meaning
masks.pp10ext	MUST be zero.
masks.newEATypeface	MUST be zero.
masks.csTypeface	MUST be zero.
masks.pp11ext	MUST be zero.

fontStyle (2 bytes): A CFStyle structure that specifies the character-level style. It MUST exist if and only if one or more of the following fields are **TRUE**: **masks.bold**, **masks.italic**, **masks.underline**, **masks.shadow**, **masks.fehint**, **masks.kumi**, **masks.emboss,** or **masks.fHasStyle**.

fontRef (2 bytes): An optional FontIndexRef that specifies the font. It MUST exist if and only if **masks.typeface** is **TRUE**.

oldEAFontRef (2 bytes): An optional FontIndexRef that specifies an East Asian font. It MUST exist if and only if **masks.oldEATypeface** is **TRUE**.

ansiFontRef (2 bytes): An optional FontIndexRef that specifies an ANSI font. It MUST exist if and only if **masks.ansiTypeface** is **TRUE**.

symbolFontRef (2 bytes): An optional FontIndexRef that specifies a symbol font. It MUST exist if and only if **masks.symbolTypeface** is **TRUE**.

fontSize (2 bytes): An optional signed integer that specifies the size, in points, of the font. It MUST be greater than or equal to 1 and less than or equal to 4000. It MUST exist if and only if **masks.size** is **TRUE**.

color (4 bytes): An optional ColorIndexStruct structure that specifies the color of the text. It MUST exist if and only if **masks.color** is **TRUE**.

position (2 bytes): An optional **signed integer** that specifies the baseline position of a text run relative to the baseline of the text line as a percentage of line height. It MUST be greater than or equal to -100 and less than or equal to 100. It MUST exist if and only if **masks.position** is **TRUE**.

2.9.15 CFMasks

Referenced by: TextCFException, TextCFException10, TextCFException9

A structure that specifies character-level font, text-formatting, and extensibility options.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
A	B	C	D	E	F	G	H	I	J	K				L		M	N	O	P	Q	R	S	T	U	V	W	reserved				

A - bold (1 bit): A bit that specifies whether the **fontStyle.bold** field of the TextCFException structure that contains this CFMasks is valid.

B - italic (1 bit): A bit that specifies whether the **fontStyle.italic** field of the TextCFException structure that contains this CFMasks is valid.

C - underline (1 bit): A bit that specifies whether the **fontStyle.underline** field of the TextCFException structure that contains this **CFMasks** is valid.

D - unused1 (1 bit): Undefined and MUST be ignored.

E - shadow (1 bit): A bit that specifies whether the **fontStyle.shadow** field of the TextCFException structure that contains this **CFMasks** is valid.

F - fehint (1 bit): A bit that specifies whether the **fontStyle.fehint** field of the TextCFException structure that contains this **CFMasks** is valid.

G - unused2 (1 bit): Undefined and MUST be ignored.

H - kumi (1 bit): A bit that specifies whether the **fontStyle.kumi** field of the TextCFException structure that contains this **CFMasks** is valid.

I - unused3 (1 bit): Undefined and MUST be ignored.

J - emboss (1 bit): A bit that specifies whether the **fontStyle.emboss** field of the TextCFException structure that contains this **CFMasks** is valid.

K - fHasStyle (4 bits): An unsigned integer that specifies whether the **fontStyle** field of the TextCFException structure that contains this **CFMasks** exists.

L - unused4 (2 bits): Undefined and MUST be ignored.

M - typeface (1 bit): A bit that specifies whether the **fontRef** field of the TextCFException structure that contains this **CFMasks** exists.

N - size (1 bit): A bit that specifies whether the **fontSize** field of the TextCFException structure that contains this **CFMasks** exists.

O - color (1 bit): A bit that specifies whether the **color** field of the TextCFException structure that contains this **CFMasks** exists.

P - position (1 bit): A bit that specifies whether the **position** field of the TextCFException structure that contains this **CFMasks** exists.

Q - pp10ext (1 bit): A bit that specifies whether the **pp10runid** and **unused** fields of the TextCFException2.9.17 structure that contains this **CFMasks** exist.

R - oldEATypeface (1 bit): A bit that specifies whether the **oldEAFontRef** field of the TextCFException structure that contains this **CFMasks** exists.

S - ansiTypeface (1 bit): A bit that specifies whether the **ansiFontRef** field of the TextCFException structure that contains this **CFMasks** exists.

T - symbolTypeface (1 bit): A bit that specifies whether the **symbolFontRef** field of the TextCFException structure that contains this **CFMasks** exists.

U - newEATypeface (1 bit): A bit that specifies whether the **newEAFontRef** field of the TextCFException2.9.18 structure that contains this **CFMasks** exists.

V - csTypeface (1 bit): A bit that specifies whether the **csFontRef** field of the TextCFException2.9.18 structure that contains this CFMasks exists.

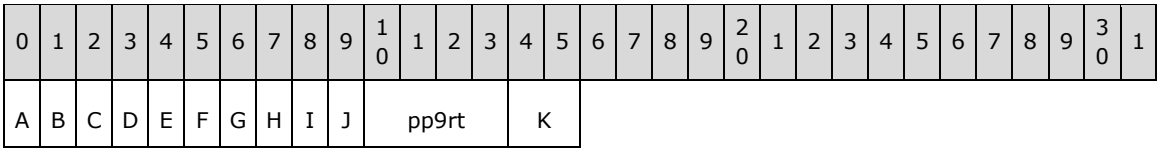
W - pp11ext (1 bit): A bit that specifies whether the **pp11ext** field of the TextCFException2.9.18 structure that contains this **CFMasks** exists.

reserved (5 bits): MUST be zero and MUST be ignored.

2.9.16 CFStyle

Referenced by: TextCFException

A structure that specifies character-level text formatting.



A - bold (1 bit): A bit that specifies whether the characters are bold.

B - italic (1 bit): A bit that specifies whether the characters are italicized.

C - underline (1 bit): A bit that specifies whether the characters are underlined.

D - unused1 (1 bit): Undefined and MUST be ignored.

E - shadow (1 bit): A bit that specifies whether the characters have a shadow effect.

F - fehint (1 bit): A bit that specifies whether characters originated from double-byte input.

G - unused2 (1 bit): Undefined and MUST be ignored.

H - kumi (1 bit): A bit that specifies whether Kumimoji are used for vertical text.

I - unused3 (1 bit): Undefined and MUST be ignored.

J - emboss (1 bit): A bit that specifies whether the characters are embossed.

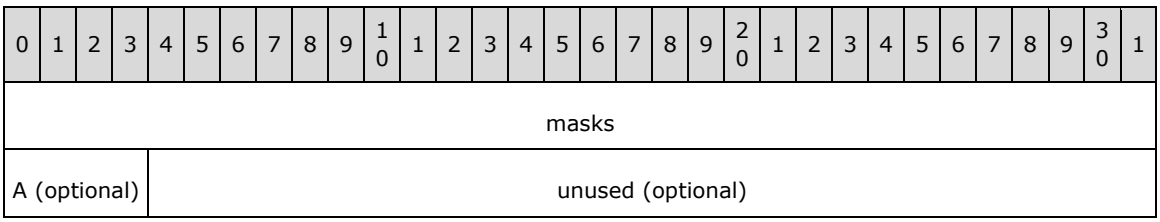
pp9rt (4 bits): An unsigned integer that specifies the run grouping of additional text properties in StyleTextProp2.9.67Atom record.

K - unused4 (2 bits): Undefined and MUST be ignored.

2.9.17 TextCFException9

Referenced by: StyleTextProp9, TextDefaults9Atom, TextMasterStyle9Level

A structure that specifies additional character-level formatting.



masks (4 bytes): A CFMasks structure that specifies which fields of this **TextCFException9** exist and are valid. Sub-fields are further specified in the following table.

Field	Meaning
masks.bold	MUST be zero.
masks.italic	MUST be zero.
masks.underline	MUST be zero.
masks.shadow	MUST be zero.

Field	Meaning
masks.fehint	MUST be zero.
masks.kumi	MUST be zero.
masks.emboss	MUST be zero.
masks.fHasStyle	MUST be zero.
masks.typeface	MUST be zero.
masks.size	MUST be zero.
masks.color	MUST be zero.
masks.position	MUST be zero.
masks.oldEATypeface	MUST be zero.
masks.ansiTypeface	MUST be zero.
masks.symbolTypeface	MUST be zero.
masks.newEATypeface	MUST be zero.
masks.csTypeface	MUST be zero.
masks.pp11ext	MUST be zero.

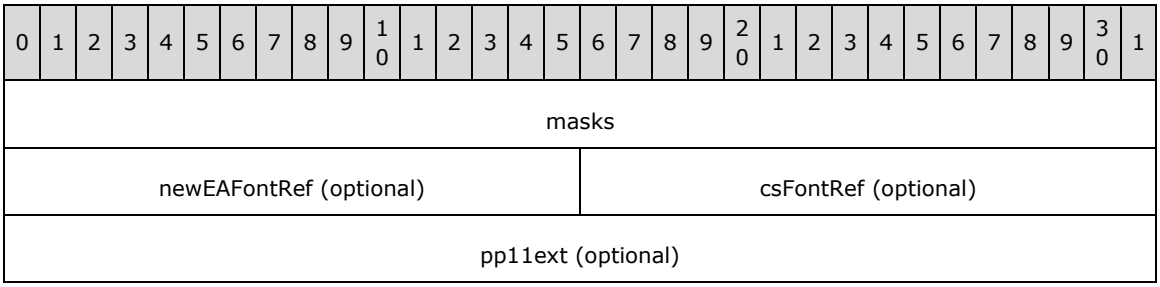
A - pp10runid (4 bits): An optional unsigned integer that specifies an identifier for a character run that contains TextCFException2.9.18 data. It MUST exist if and only if **masks.pp10ext** is **TRUE**.

unused (28 bits): Undefined and MUST be ignored. It MUST exist if and only if **masks.pp10ext** is **TRUE**.

2.9.18 TextCFException10

Referenced by: StyleTextProp10Atom, TextDefaults10Atom, TextMasterStyle10Level

A structure that specifies additional character-level formatting information.



masks (4 bytes): A CFMasks structure that specifies which fields of this **TextCFException10** exist and are valid.

Field	Meaning
masks.bold	MUST be zero.
masks.italic	MUST be zero.
masks.underline	MUST be zero.
masks.shadow	MUST be zero.
masks.fehint	MUST be zero.
masks.kumi	MUST be zero.
masks.emboss	MUST be zero.
masks.fHasStyle	MUST be zero.
masks.typeface	MUST be zero.

Field	Meaning
masks.size	MUST be zero.
masks.color	MUST be zero.
masks.position	MUST be zero.
masks.pp10ext	MUST be zero.
masks.oldEATypeface	MUST be zero.
masks.ansiTypeface	MUST be zero.
masks.symbolTypeface	MUST be zero.

newEAFontRef (2 bytes): An optional FontIndexRef2.2.11 that specifies a new East Asian font. It MUST exist if and only if **masks.newEATypeface** is **TRUE**.

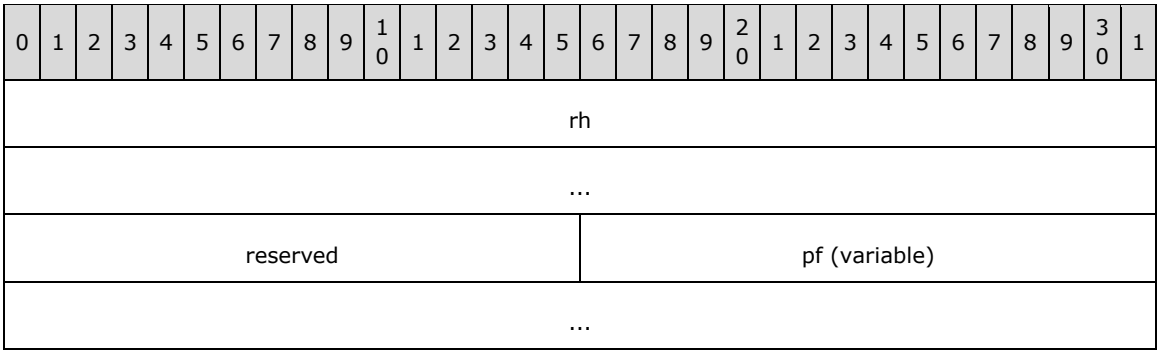
csFontRef (2 bytes): An optional FontIndexRef2.2.11 that specifies a complex script font. It MUST exist if and only if **masks.csTypeface** is **TRUE**.

pp11ext (4 bytes): Undefined and MUST be ignored. It MUST exist if and only if **masks.pp11ext** is **TRUE**.

2.9.19 TextPFExceptionAtom

Referenced by: DocumentTextInfoContainer

An atom record that specifies paragraph-level formatting.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TextParagraphFormatExceptionAtom.

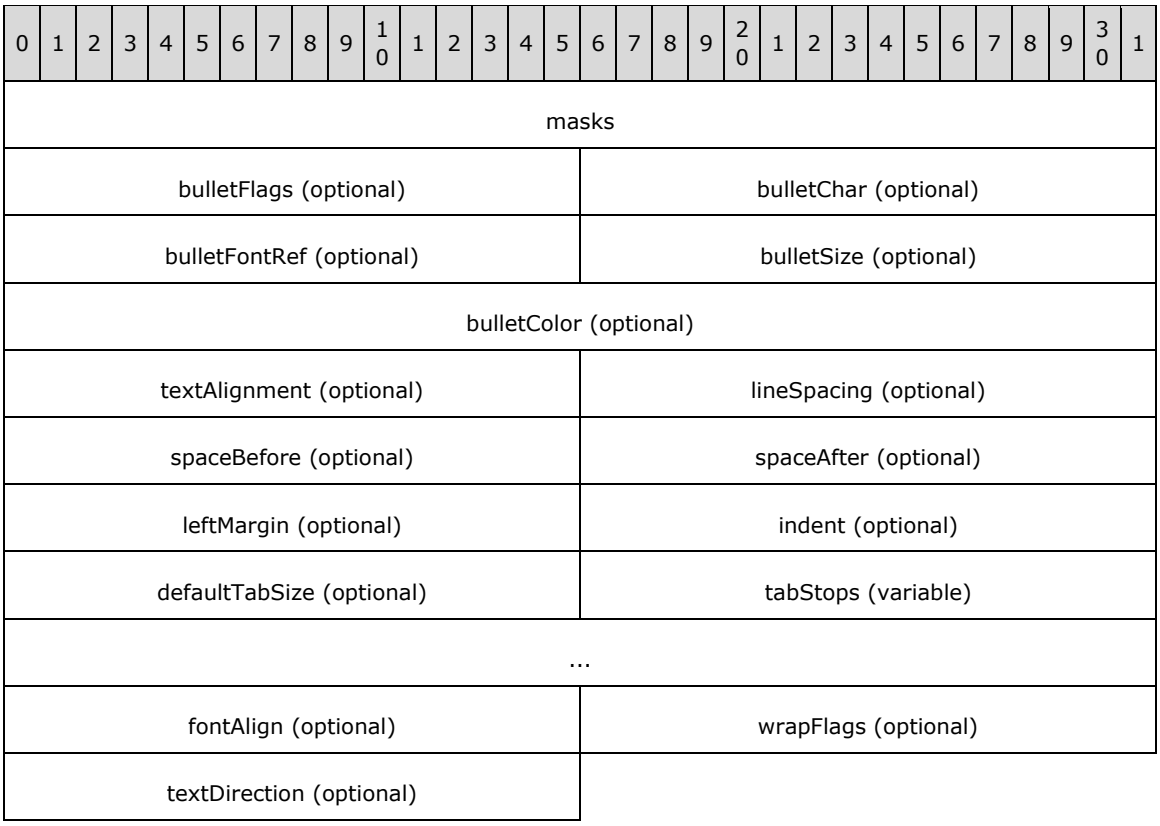
reserved (2 bytes): MUST be zero and MUST be ignored.

pf (variable): A TextPFException structure that specifies paragraph-level formatting.

2.9.20 TextPFException

Referenced by: TextMasterStyleLevel, TextPFExceptionAtom, TextPFRun

A structure that specifies paragraph-level formatting.



masks (4 bytes): A PFMasks structure that specifies whether certain fields of this **TextPFException** record exist and are valid. Sub-fields are further specified in the following table.

Field	Meaning
masks.bulletBlip	MUST be zero.
masks.bulletHasScheme	MUST be zero.
masks.bulletScheme	MUST be zero.

bulletFlags (2 bytes): An optional BulletFlags structure that specifies whether certain bullet-related fields are valid. It MUST exist if and only if any of **masks.hasBullet**, **masks.bulletHasFont**, **masks.bulletHasColor**, or **masks.bulletHasSize** is **TRUE**.

bulletChar (2 bytes): An optional **signed integer** that specifies a **UTF-16 Unicode** [RFC2781] character to display as the bullet. The character MUST NOT be the NUL character 0x0000. It MUST exist if and only if **masks.bulletChar** is **TRUE**.

bulletFontRef (2 bytes): An optional FontIndexRef that specifies the font to use for the bullet. It MUST exist if and only if **masks.bulletFont** is **TRUE**. This field is valid if and only if **bulletFlags.fBulletHasFont** is **TRUE**.

bulletSize (2 bytes): An optional BulletSize that specifies the size of the bullet. It MUST exist if and only if **masks.bulletSize** is **TRUE**. This field is valid if and only if **bulletFlags.fBulletHasSize** is **TRUE**.

bulletColor (4 bytes): An optional ColorIndexStruct structure that specifies the color of a bullet. This field exists if and only if **masks.bulletColor** is **TRUE**. This field is valid if and only if **bulletFlags.fBulletHasColor** is **TRUE**.

textAlignment (2 bytes): An optional **TextAlignmentEnum** enumeration that specifies the alignment of the paragraph. It MUST exist if and only if **masks.align** is **TRUE**.

lineSpacing (2 bytes): An optional ParaSpacing that specifies the spacing between lines in the paragraph. It MUST exist if and only if **masks.lineSpacing** is **TRUE**.

spaceBefore (2 bytes): An optional ParaSpacing that specifies the size of the spacing before the paragraph. It MUST exist if and only if **masks.spaceBefore** is **TRUE**.

spaceAfter (2 bytes): An optional ParaSpacing that specifies the size of the spacing after the paragraph. It MUST exist if and only if **masks.spaceAfter** is **TRUE**.

leftMargin (2 bytes): An optional MarginOrIndent that specifies the left margin of the paragraph. It MUST exist if and only if **masks.leftMargin** is **TRUE**.

indent (2 bytes): An optional MarginOrIndent that specifies the indentation of the paragraph. It MUST exist if and only if **masks.indent** is **TRUE**.

defaultTabSize (2 bytes): An optional TabSize that specifies the default tab size of the paragraph. It MUST exist if and only if **masks.defaultTabSize** is **TRUE**.

tabStops (variable): An optional TabStops structure that specifies the tab stops for the paragraph. It MUST exist if and only if **masks.tabStops** is **TRUE**.

fontAlign (2 bytes): An optional **TextFontAlignmentEnum** enumeration that specifies the font alignment of the text in the paragraph. It MUST exist if and only if **masks.fontAlign** is **TRUE**.

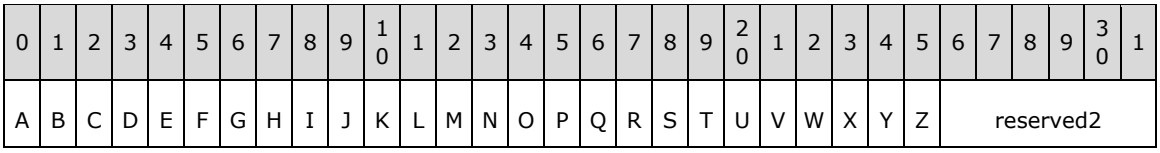
wrapFlags (2 bytes): An optional PFWrapFlags structure that specifies text-wrapping options for the paragraph. It MUST exist if and only if any of **masks.charWrap**, **masks.wordWrap**, or **masks.overflow** is **TRUE**.

textDirection (2 bytes): An optional **TextDirectionEnum** enumeration that specifies the direction of the text in this paragraph. It MUST exist if and only if **masks.textDirection** is **TRUE**.

2.9.21 PFMasks

Referenced by: TextPFException, TextPFException9

A structure that specifies which paragraph-level formatting fields are valid in the TextPFException or TextPFException2.9.26 record that contains this **PFMasks** structure.



A - hasBullet (1 bit): A bit that specifies whether the **bulletFlags** field of the TextPFException structure that contains this **PFMasks** exists and whether **bulletFlags.fHasBullet** is valid.

B - bulletHasFont (1 bit): A bit that specifies whether the **bulletFlags** field of the TextPFException structure that contains this **PFMasks** exists and whether **bulletFlags.fBulletHasFont** is valid.

C - bulletHasColor (1 bit): A bit that specifies whether the **bulletFlags** field of the TextPFException structure that contains this **PFMasks** exists and whether **bulletFlags.fBulletHasColor** is valid.

D - bulletHasSize (1 bit): A bit that specifies whether the **bulletFlags** field of the TextPFException structure that contains this **PFMasks** exists and whether **bulletFlags.fBulletHasSize** is valid.

E - bulletFont (1 bit): A bit that specifies whether the **bulletFontRef** field of the TextPFException structure that contains this **PFMasks** exists.

F - bulletColor (1 bit): A bit that specifies whether the **bulletColor** field of the TextPFException structure that contains this **PFMasks** exists.

G - bulletSize (1 bit): A bit that specifies whether the **bulletSize** field of the TextPFException structure that contains this **PFMasks** exists.

H - bulletChar (1 bit): A bit that specifies whether the **bulletChar** field of the TextPFException structure that contains this **PFMasks** exists.

I - leftMargin (1 bit): A bit that specifies whether the **leftMargin** field of the TextPFException structure that contains this **PFMasks** exists.

J - unused (1 bit): Undefined and MUST be ignored.

K - indent (1 bit): A bit that specifies whether the **indent** field of the TextPFException structure that contains this **PFMasks** exists.

L - align (1 bit): A bit that specifies whether the **textAlignment** field of the TextPFException structure that contains this **PFMasks** exists.

M - lineSpacing (1 bit): A bit that specifies whether the **lineSpacing** field of the TextPFException structure that contains this **PFMasks** exists.

N - spaceBefore (1 bit): A bit that specifies whether the **spaceBefore** field of the TextPFException that contains this **PFMasks** exists.

O - spaceAfter (1 bit): A bit that specifies whether the **spaceAfter** field of the TextPFException structure that contains this **PFMasks** exists.

P - defaultTabSize (1 bit): A bit that specifies whether the **defaultTabSize** field of the TextPFException structure that contains this **PFMasks** exists.

Q - fontAlign (1 bit): A bit that specifies whether the **fontAlign** field of the TextPFException structure that contains this **PFMasks** exists.

R - charWrap (1 bit): A bit that specifies whether the **wrapFlags** field of the TextPFException structure that contains this PFMasks exists and whether **wrapFlags.charWrap** is valid.

S - wordWrap (1 bit): A bit that specifies whether the **wrapFlags** field of the TextPFException structure that contains this **PFMasks** exists and whether **wrapFlags.wordWrap** is valid.

T - overflow (1 bit): A bit that specifies whether the **wrapFlags** field of the TextPFException structure that contains this **PFMasks** exists and whether **wrapFlags.overflow** is valid.

U - tabStops (1 bit): A bit that specifies whether the **tabStops** field of the TextPFException structure that contains this **PFMasks** exists.

V - textDirection (1 bit): A bit that specifies whether the **textDirection** field of the TextPFException structure that contains this **PFMasks** exists.

W - reserved1 (1 bit): MUST be zero and MUST be ignored.

X - bulletBlip (1 bit): A bit that specifies whether the **bulletBlipRef** field of the TextPFException2.9.26 structure that contains this **PFMasks** exists.

Y - bulletScheme (1 bit): A bit that specifies whether the **bulletAutoNumberScheme** field of the TextPFException2.9.26 structure that contains this **PFMasks** exists.

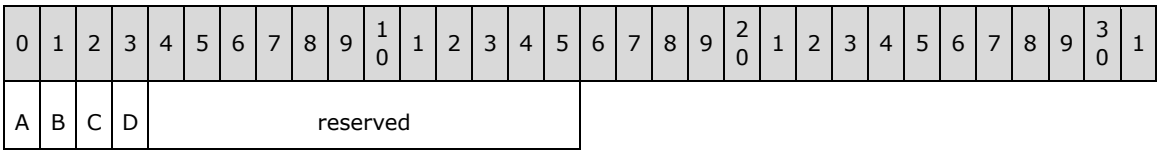
Z - bulletHasScheme (1 bit): A bit that specifies whether the **fBulletHasAutoNumber** field of the TextPFException2.9.26 structure that contains this **PFMasks** exists.

reserved2 (6 bits): MUST be zero and MUST be ignored.

2.9.22 BulletFlags

Referenced by: TextPFException

A structure that specifies bullet properties.



A - fHasBullet (1 bit): A **bit** that specifies whether a bullet exists.

B - fBulletHasFont (1 bit): A bit that specifies whether the bullet has a font.

C - fBulletHasColor (1 bit): A bit that specifies whether the bullet has a color.

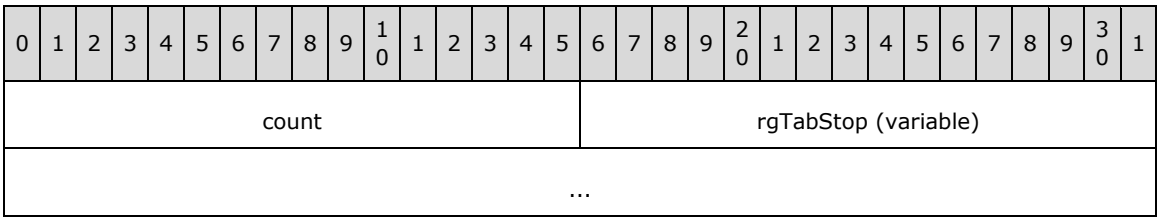
D - fBulletHasSize (1 bit): A bit that specifies whether the bullet has a size.

reserved (12 bits): MUST be zero and MUST be ignored.

2.9.23 TabStops

Referenced by: TextPFException, TextRuler

A structure that specifies tab stops.



count (2 bytes): An unsigned integer that specifies the count of tab stops in **rgTabStop**.

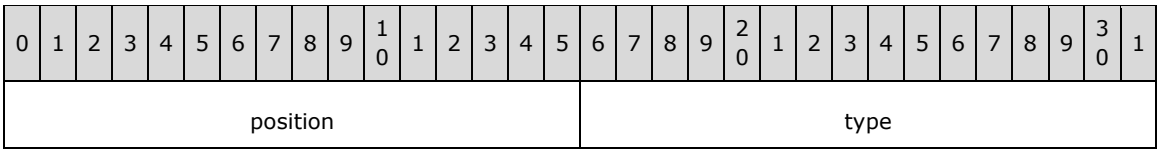
rgTabStop (variable): An array of TabStop that specifies the tab stops. The length, in bytes, of the array is specified by the following formula:

```
count * 4
```

2.9.24 TabStop

Referenced by: TabStops

A structure that specifies a tab stop.



position (2 bytes): A **signed integer** that specifies an offset, in master units, of the tab stop.

If the TextPFException record that contains this **TabStop** structure also contains a **leftMargin**, then the value of **position** is relative to the left margin of the paragraph; otherwise, the value is relative to the left side of the paragraph.

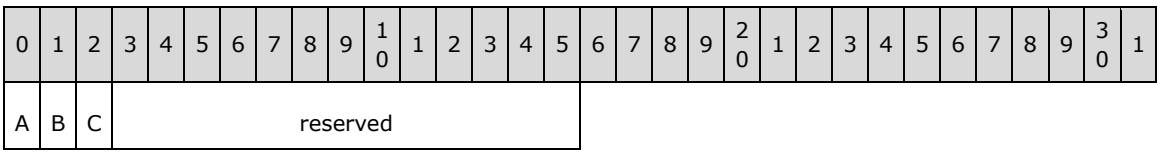
If a TextRuler record contains this **TabStop** structure, the value is relative to the left side of the text ruler.

type (2 bytes): A **TextTabTypeEnum** enumeration that specifies how text aligns at the tab stop.

2.9.25 PFWrapFlags

Referenced by: TextPFException

A structure that specifies line breaking settings for a paragraph.



A - charWrap (1 bit): A bit that specifies whether the paragraph follows the East Asian text line breaking settings specified by the **KinsokuContainer** (section 2.9.2) and **Kinsoku9Container** (section 2.9.6) records.

B - wordWrap (1 bit): A bit that specifies whether text wraps from one line to the next only at word breaks, or in the middle of a word. It MUST be a value from the following table.

Value	Meaning
TRUE	Text wraps at word breaks. All of the characters of a word exist on the same line.
FALSE	Text wraps at character breaks. Characters of a word may be split across lines.

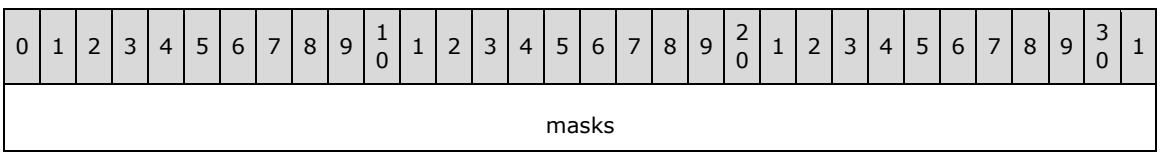
C - overflow (1 bit): A bit that specifies whether hanging punctuation is allowed for East Asian text.

reserved (13 bits): MUST be zero and MUST be ignored.

2.9.26 TextPFException9

Referenced by: StyleTextProp9, TextDefaults9Atom, TextMasterStyle9Level

A structure that specifies additional paragraph-level formatting.



bulletBlipRef (optional)	fBulletHasAutoNumber (optional)
bulletAutoNumberScheme (optional)	

masks (4 bytes): A PFMasks structure that specifies which fields in this **TextPFException9** exist. Sub-fields are further specified in the following table.

Field	Meaning
masks.hasBullet	MUST be zero.
masks.bulletHasFont	MUST be zero.
masks.bulletHasColor	MUST be zero.
masks.bulletHasSize	MUST be zero.
masks.bulletFont	MUST be zero.
masks.bulletColor	MUST be zero.
masks.bulletSize	MUST be zero.
masks.bulletChar	MUST be zero.
masks.leftMargin	MUST be zero.
masks.indent	MUST be zero.
masks.align	MUST be zero.
masks.lineSpacing	MUST be zero.
masks.spaceBefore	MUST be zero.
masks.spaceAfter	MUST be zero.
masks.defaultTabSize	MUST be zero.
masks.fontAlign	MUST be zero.
masks.charWrap	MUST be zero.
masks.wordWrap	MUST be zero.
masks.overflow	MUST be zero.
masks.tabStops	MUST be zero.
masks.textDirection	MUST be zero.

bulletBlipRef (2 bytes): An optional BlipRef that specifies a picture to use as a bullet for this paragraph. It MUST exist if and only if **masks.bulletBlip** is **TRUE**.

fBulletHasAutoNumber (2 bytes): An optional **signed integer** that specifies whether this paragraph has an automatic numbering scheme. It MUST exist if and only if **masks.bulletHasScheme** is **TRUE**. It MUST be a value from the following table.

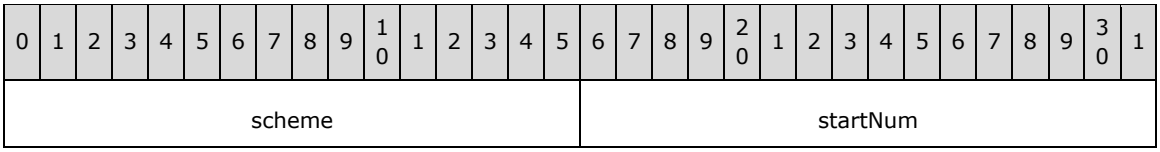
Field	Meaning
0x0000	This paragraph does not have an automatic numbering scheme.
0x0001	This paragraph has an automatic numbering scheme.

bulletAutoNumberScheme (4 bytes): An optional TextAutoNumberScheme structure that specifies the automatic numbering scheme for this paragraph. It MUST exist if and only if **masks.bulletScheme** is **TRUE**.

2.9.27 TextAutoNumberScheme

Referenced by: TextPFException9

A structure that specifies the automatic numbering scheme for text paragraphs.



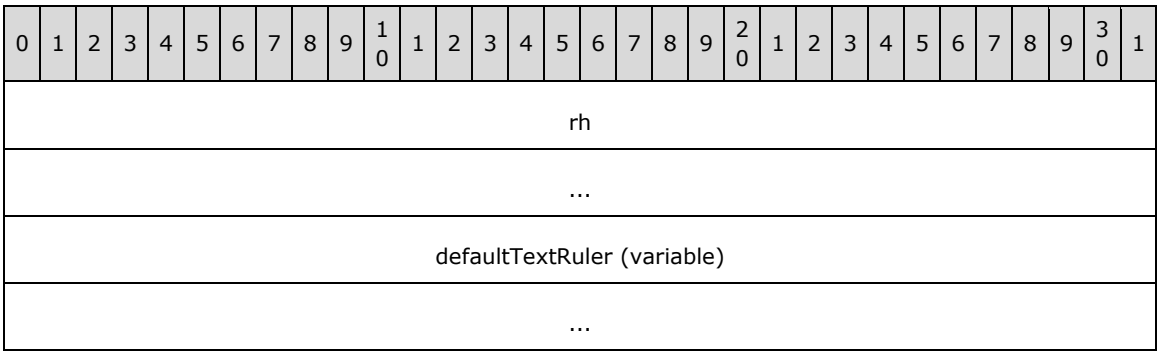
scheme (2 bytes): A **TextAutoNumberSchemeEnum** enumeration that specifies the scheme. The scheme describes the style of the number bullets.

startNum (2 bytes): A **signed integer** that specifies the numeric value of the first number assigned. It MUST be greater than or equal to 0x0001.

2.9.28 DefaultRulerAtom

Referenced by: DocumentTextInfoContainer

An atom record that specifies the default ruler and corresponding options.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_DefaultRulerAtom.

defaultTextRuler (variable): A TextRuler structure that specifies the default text ruler. Sub-fields are further specified in the following table.

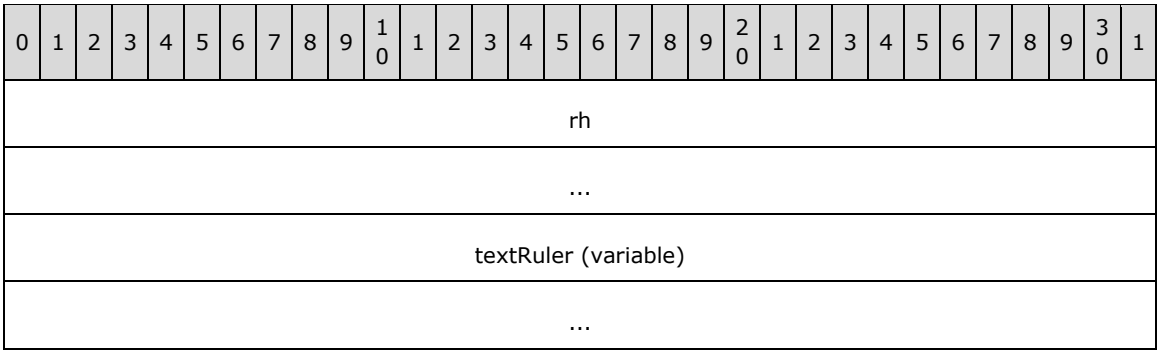
Field	Meaning
defaultTextRuler.fDefaultTabSize	MUST be **TRUE**.
defaultTextRuler.fCLevels	MUST be **TRUE**.
defaultTextRuler.fTabStops	MUST be **TRUE**.
defaultTextRuler.fLeftMargin1	MUST be **TRUE**.
defaultTextRuler.fLeftMargin2	MUST be **TRUE**.
defaultTextRuler.fLeftMargin3	MUST be **TRUE**.

Field	Meaning
defaultTextRuler.fLeftMargin4	MUST be **TRUE**.
defaultTextRuler.fLeftMargin5	MUST be **TRUE**.
defaultTextRuler.fIndent1	MUST be **TRUE**.
defaultTextRuler.fIndent2	MUST be **TRUE**.
defaultTextRuler.fIndent3	MUST be **TRUE**.
defaultTextRuler.fIndent4	MUST be **TRUE**.
defaultTextRuler.fIndent5	MUST be **TRUE**.

2.9.29 TextRulerAtom

Referenced by: TextClientDataSubContainerOrAtom

An atom record that specifies a **text ruler**.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

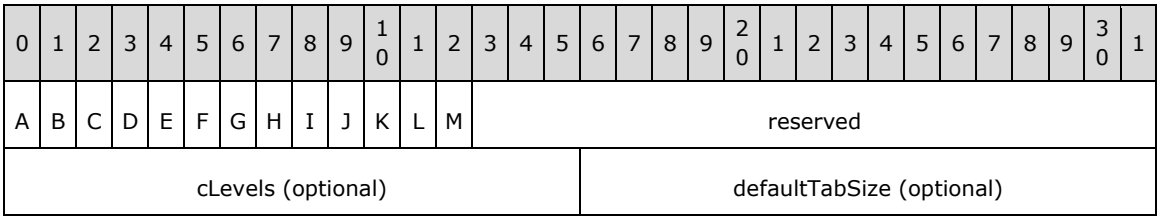
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TextRulerAtom.

textRuler (variable): A TextRuler structure that specifies a text ruler.

2.9.30 TextRuler

Referenced by: DefaultRulerAtom, TextRulerAtom

A structure that specifies tabbing, margins, and indentation for text.



tabs (variable)	
...	
leftMargin1 (optional)	indent1 (optional)
leftMargin2 (optional)	indent2 (optional)
leftMargin3 (optional)	indent3 (optional)
leftMargin4 (optional)	indent4 (optional)
leftMargin5 (optional)	indent5 (optional)

A - fDefaultTabSize (1 bit): A bit that specifies whether **defaultTabSize** exists.

B - fCLevels (1 bit): A bit that specifies whether **cLevels** exists.

C - fTabStops (1 bit): A bit that specifies whether **tabs** exists.

D - fLeftMargin1 (1 bit): A bit that specifies whether **leftMargin1** exists.

E - fLeftMargin2 (1 bit): A bit that specifies whether **leftMargin2** exists.

F - fLeftMargin3 (1 bit): A bit that specifies whether **leftMargin3** exists.

G - fLeftMargin4 (1 bit): A bit that specifies whether **leftMargin4** exists.

H - fLeftMargin5 (1 bit): A bit that specifies whether **leftMargin5** exists.

I - fIndent1 (1 bit): A bit that specifies whether **indent1** exists.

J - fIndent2 (1 bit): A bit that specifies whether **indent2** exists.

K - fIndent3 (1 bit): A bit that specifies whether **indent3** exists.

L - fIndent4 (1 bit): A bit that specifies whether **indent4** exists.

M - fIndent5 (1 bit): A bit that specifies whether **indent5** exists.

reserved (19 bits): MUST be zero and MUST be ignored.

cLevels (2 bytes): An optional signed integer that specifies the number of style levels in this text ruler. It MUST exist if and only if **fCLevels** is **TRUE**.

defaultTabSize (2 bytes): An optional TabSize that specifies the default tab size for this text ruler. It MUST exist if and only if **fDefaultTabSize** is **TRUE**.

tabs (variable): An optional TabStops structure that specifies the tab stops for this text ruler. It MUST exist if and only if **fTabStops** is **TRUE**.

leftMargin1 (2 bytes): An optional MarginOrIndent that specifies the left margin for text that has an IndentLevel equal to 0x0000. It MUST exist if and only if **fLeftMargin1** is **TRUE**.

indent1 (2 bytes): An optional MarginOrIndent that specifies the indentation for text that has an IndentLevel equal to 0x0000. It MUST exist if and only if **fIndent1** is **TRUE**.

leftMargin2 (2 bytes): An optional MarginOrIndent that specifies the left margin for text that has an IndentLevel equal to 0x0001. It MUST exist if and only if **fLeftMargin2** is **TRUE**.

indent2 (2 bytes): An optional MarginOrIndent that specifies the indentation for text that has an IndentLevel equal to 0x0001. It MUST exist if and only if **fIndent2** is **TRUE**.

leftMargin3 (2 bytes): An optional MarginOrIndent that specifies the left margin for text that has an IndentLevel equal to 0x0002. It MUST exist if and only if **fLeftMargin3** is **TRUE**.

indent3 (2 bytes): An optional MarginOrIndent that specifies the indentation for text that has an IndentLevel equal to 0x0002. It MUST exist if and only if **fIndent3** is **TRUE**.

leftMargin4 (2 bytes): An optional MarginOrIndent that specifies the left margin for text that has an IndentLevel equal to 0x0003. It MUST exist if and only if **fLeftMargin4** is **TRUE**.

indent4 (2 bytes): An optional MarginOrIndent that specifies the indentation for text that has an IndentLevel equal to 0x0003. It MUST exist if and only if **fIndent4** is **TRUE**.

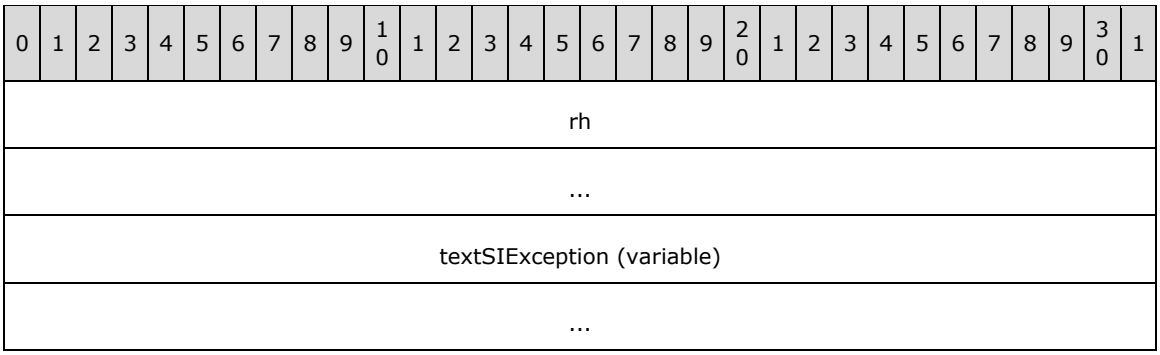
leftMargin5 (2 bytes): An optional MarginOrIndent that specifies the left margin for text that has an IndentLevel equal to 0x0004. It MUST exist if and only if **fLeftMargin5** is **TRUE**.

indent5 (2 bytes): An optional MarginOrIndent that specifies the indentation for text that has an IndentLevel equal to 0x0004. It MUST exist if and only if **fIndent5** is **TRUE**.

2.9.31 TextSIExceptionAtom

Referenced by: DocumentTextInfoContainer

An atom record that specifies default settings for language and spelling information.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TextSpecialInfoDefaultAtom.

textSIException (variable): A TextSIException structure that specifies default settings for language and spelling information. The length, in bytes, of the field is specified by **rh.recLen**. Sub-fields are further specified in the following table.

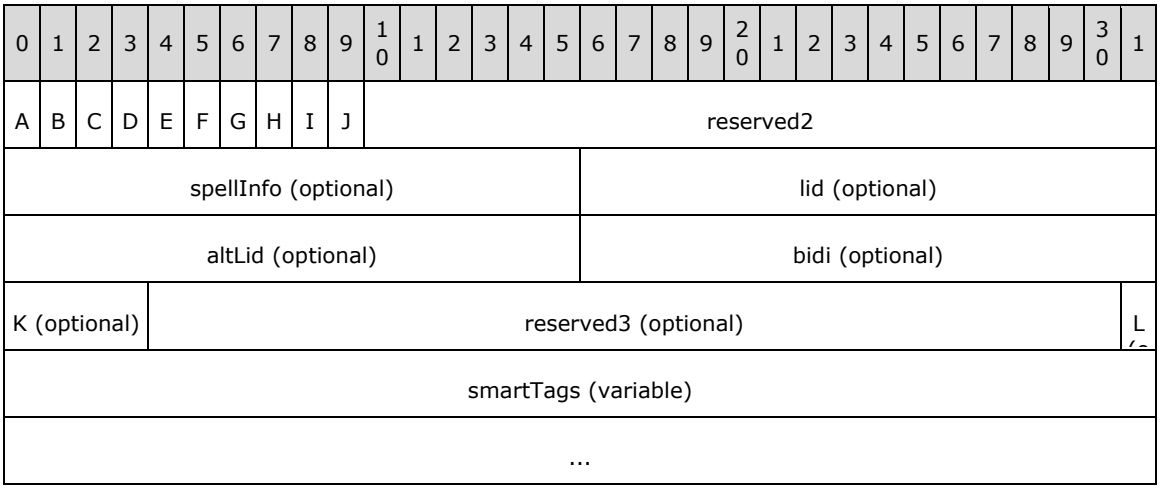
Field	Meaning
textSIException.fPp10ext	MUST be zero.

Field	Meaning
textSIException.fBidi	MUST be zero.
textSIException.smartTag	MUST be zero.

2.9.32 TextSIException

Referenced by: StyleTextProp11, StyleTextProp9, TextSIExceptionAtom, TextSIRun

A structure that specifies additional text properties.



A - spell (1 bit): A bit that specifies whether the **spellInfo** field exists.

B - lang (1 bit): A bit that specifies whether the **lid** field exists.

C - altLang (1 bit): A bit that specifies whether the **altLid** field exists.

D - unused1 (1 bit): Undefined and MUST be ignored.

E - unused2 (1 bit): Undefined and MUST be ignored.

F - fPp10ext (1 bit): A bit that specifies whether the **pp10runid**, **reserved3**, and **grammarError** fields exist**.**

G - fBidi (1 bit): A bit that specifies whether the **bidi** field exists.

H - unused3 (1 bit): Undefined and MUST be ignored.

I - reserved1 (1 bit): MUST be zero and MUST be ignored.

J - smartTag (1 bit): A bit that specifies whether the **smartTags** field exists.

reserved2 (22 bits): MUST be zero and MUST be ignored.

spellInfo (2 bytes): An optional SpellingFlags structure that specifies the spelling status of this text. It MUST exist if and only if **spell** is **TRUE**. The **spellInfo.grammar** sub-field MUST be zero.

lid (2 bytes): An optional TxLCID that specifies the language identifier of this text. It MUST exist if and only if **lang** is **TRUE**.

altLid (2 bytes): An optional TxLCID that specifies the alternate language identifier of this text. It MUST exist if and only if **altLang** is **TRUE**.

bidi (2 bytes): An optional signed integer that specifies whether the text contains bidirectional characters. It MUST exist if and only if **fBidi** is **TRUE**. It MUST be a value from the following table:

Value	Meaning
0x0000	Contains no bidirectional characters.
0x0001	Contains bidirectional characters.

K - pp10runid (4 bits): An optional unsigned integer that specifies an identifier for a character run that contains StyleTextProp2.9.71 data. It MUST exist if and only if **fPp10ext** is **TRUE**.

reserved3 (27 bits): An optional unsigned integer that MUST be zero, and MUST be ignored. It MUST exist if and only if **fPp10ext** is **TRUE**.

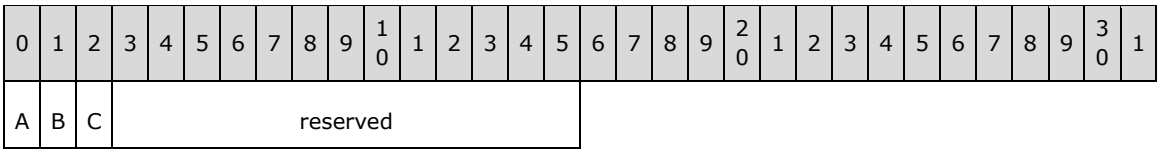
L - grammarError (1 bit): An optional bit that specifies a grammar error. It MUST exist if and only if **fPp10ext** is **TRUE**.

smartTags (variable): An optional SmartTags structure that specifies smart tags applied to the text. It MUST exist if and only if **smartTag** is **TRUE**.

2.9.33 SpellingFlags

Referenced by: TextSIException

A structure that specifies the spelling status of a run of text.



A - error (1 bit): A bit that specifies whether the text is spelled incorrectly.

B - clean (1 bit): A bit that specifies whether the text needs rechecking.

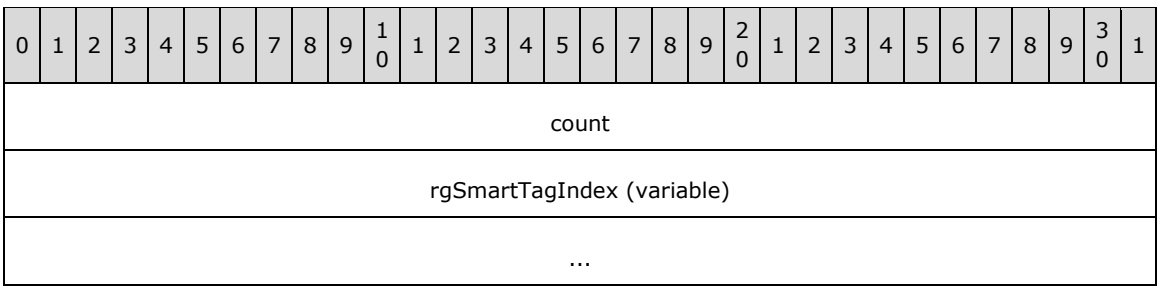
C - grammar (1 bit): A bit that specifies whether the text has a grammar error.

reserved (13 bits): MUST be zero and MUST be ignored.

2.9.34 SmartTags

Referenced by: TextSIException

A structure that specifies the smart tags attached to a run of text.



count (4 bytes): An unsigned integer specifies the count of items in **rgSmartTagIndex.**

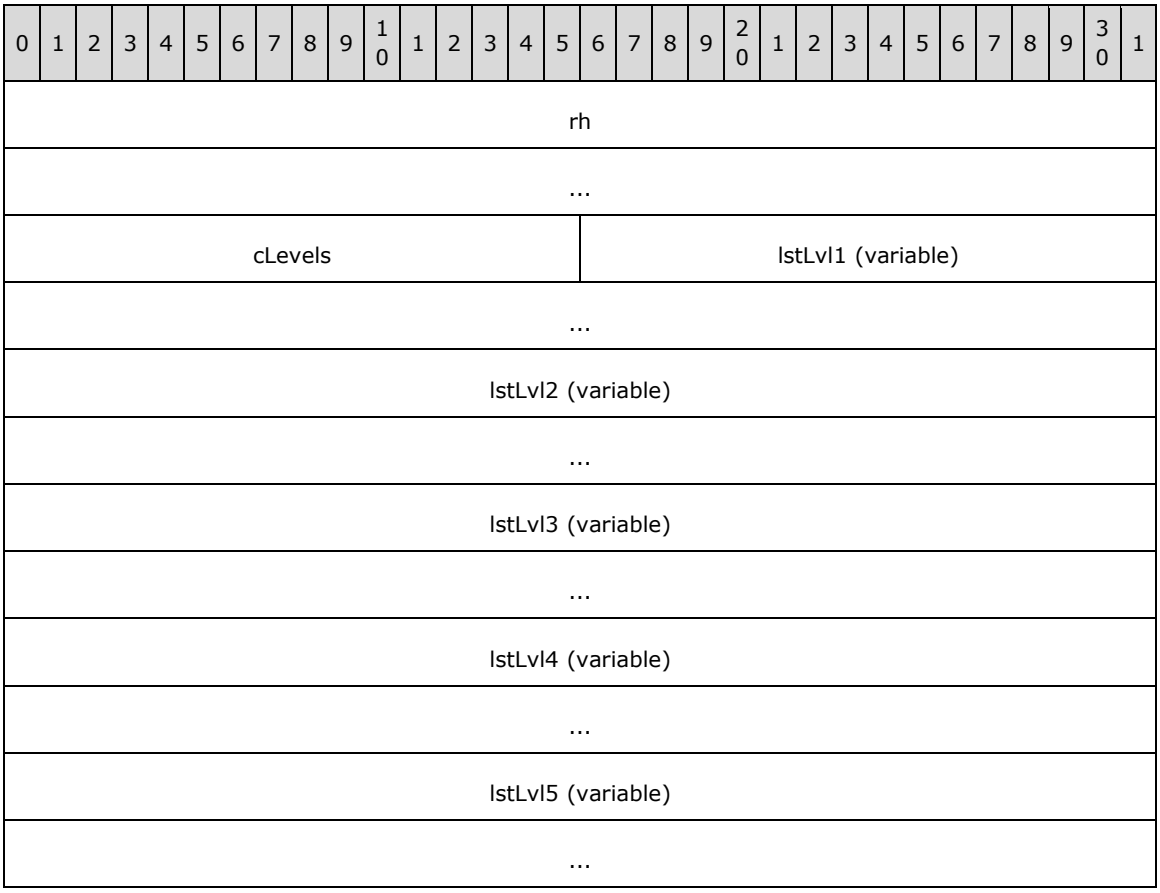
rgSmartTagIndex (variable): An array of SmartTagIndex that specifies the indices. The count of items in the array is specified by **count**.

2.9.35 TextMasterStyleAtom

Referenced by: DocumentTextInfoContainer, MainMasterContainer

An atom record that specifies the character-level and paragraph-level formatting of a main master slide.

If this **TextMasterStyleAtom** is contained in a **MainMasterContainer** record (section 2.5.3), character-level and paragraph-level formatting not specified by this **TextMasterStyleAtom** record inherit from the **TextMasterStyleAtom** record contained in the **DocumentTextInfoContainer** record (section 2.9.1).



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	Specifies the type of text to which the formatting applies. It MUST be a **TextTypeEnum** enumeration value.
rh.recType	MUST be an RT_TextMasterStyleAtom.

cLevels (2 bytes): An **unsigned integer** that specifies the number of style levels. It MUST be less than or equal to 0x0005.

lstLvl1 (variable): An optional TextMasterStyleLevel structure that specifies the master formatting for text that has an IndentLevel equal to 0x0000. It MUST exist if and only if **cLevels** is greater than 0x0000.

lstLvl2 (variable): An optional TextMasterStyleLevel structure that specifies the master formatting for text that has an IndentLevel equal to 0x0001. It MUST exist if and only if **cLevels** is greater than 0x0001.

lstLvl3 (variable): An optional TextMasterStyleLevel structure that specifies the master formatting for text that has an IndentLevel equal to 0x0002. It MUST exist if and only if **cLevels** is greater than 0x0002.

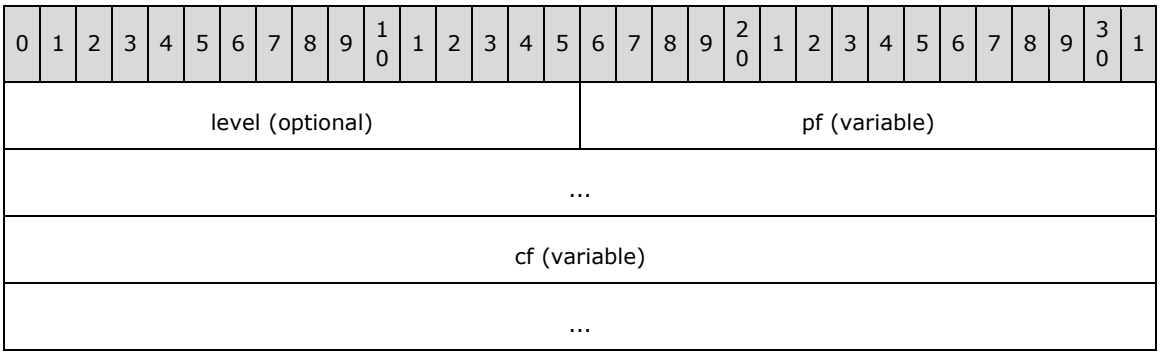
lstLvl4 (variable): An optional TextMasterStyleLevel structure that specifies the master formatting for text that has an IndentLevel equal to 0x0003. It MUST exist if and only if **cLevels** is greater than 0x0003.

lstLvl5 (variable): An optional TextMasterStyleLevel structure that specifies the master formatting for text that has an IndentLevel equal to 0x0004. It MUST exist if and only if **cLevels** is greater than 0x0004.

2.9.36 TextMasterStyleLevel

Referenced by: TextMasterStyleAtom

A structure that specifies character-level and paragraph-level formatting for a style level.



level (2 bytes): An optional **unsigned integer** that specifies to what style level this **TextMasterStyleLevel** applies. This field MUST exist if and only if the **rh.recInstance** field of the TextMasterStyleAtom record that contains this **TextMasterStyleLevel** structure is greater than or equal to 0x005. If the field exists, its value MUST be less than the **cLevels** field of the TextMasterStyleAtom record that contains this **TextMasterStyleLevel** structure.

pf (variable): A TextPFException structure that specifies paragraph-level formatting.

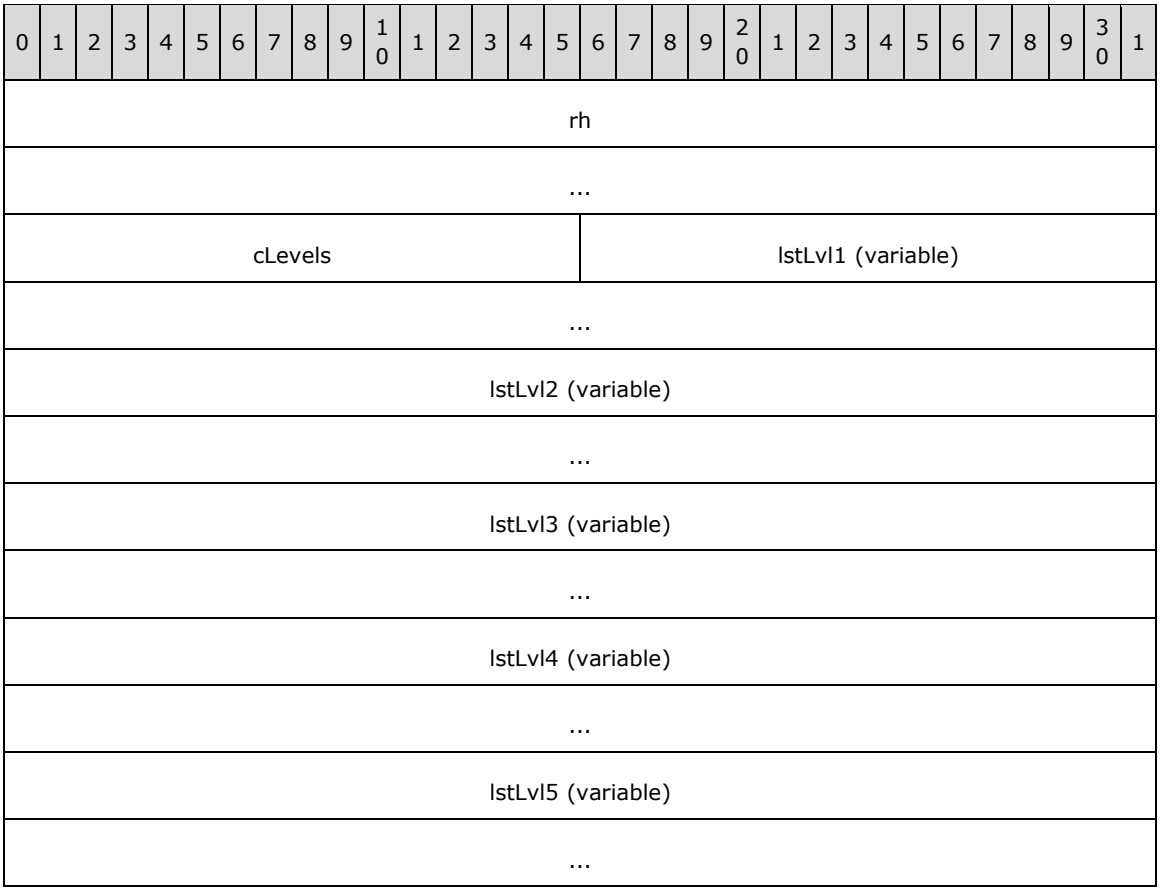
cf (variable): A TextCFException structure that specifies character-level formatting.

2.9.37 TextMasterStyle9Atom

Referenced by: PP9DocBinaryTagExtension, PP9SlideBinaryTagExtension

An atom record that specifies additional character-level and paragraph-level formatting of a main master slide.

If this **TextMasterStyle9Atom** is contained in a **MainMasterContainer** record (section 2.5.3), character-level and paragraph-level formatting not specified by this **TextMasterStyle9Atom** record inherit from the **TextMasterStyle9Atom** records contained in the PP2.4.23.5DocBinaryTagExtension record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	Specifies the type of text to which the formatting applies. It MUST be a **TextTypeEnum** enumeration value.
rh.recType	MUST be an RT_TextMasterStyle9Atom.

cLevels (2 bytes): An **unsigned integer** that specifies the number of style levels. It MUST be less than or equal to 0x0005.

lstLvl1 (variable): An optional TextMasterStyle2.9.38Level structure that specifies additional master formatting for text that has an IndentLevel equal to 0x0000. It MUST exist if and only if **cLevels** is greater than 0x0000.

lstLvl2 (variable): An optional TextMasterStyle2.9.38Level structure that specifies additional master formatting for text that has an IndentLevel equal to 0x0001. It MUST exist if and only if **cLevels** is greater than 0x0001.

lstLvl3 (variable): An optional TextMasterStyle2.9.38Level structure that specifies additional master formatting for text that has an IndentLevel equal to 0x0002. It MUST exist if and only if **cLevels** is greater than 0x0002.

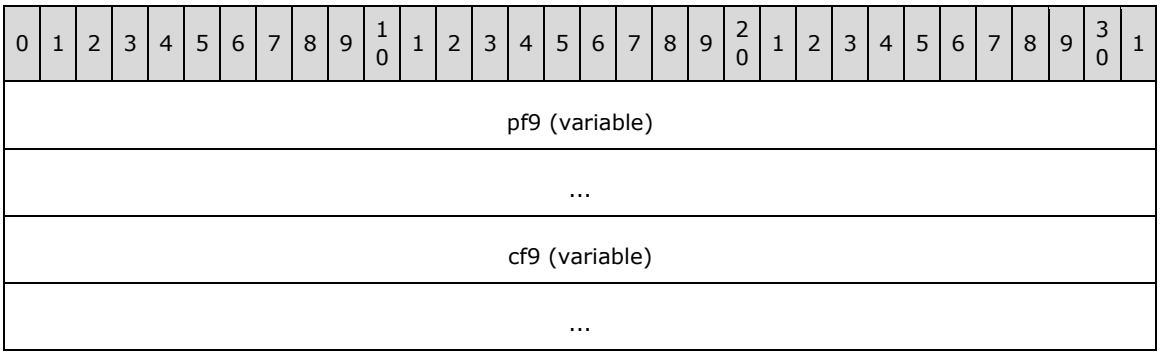
lstLvl4 (variable): An optional TextMasterStyle2.9.38Level structure that specifies additional master formatting for text that has an IndentLevel equal to 0x0003. It MUST exist if and only if **cLevels** is greater than 0x0003.

lstLvl5 (variable): An optional TextMasterStyle2.9.38Level structure that specifies additional master formatting for text that has an IndentLevel equal to 0x0004. It MUST exist if and only if **cLevels** is greater than 0x0004.

2.9.38 TextMasterStyle9Level

Referenced by: TextMasterStyle9Atom

A structure that specifies additional character-level and paragraph level-formatting for a style level.



pf9 (variable): A TextPFException2.9.26 structure that specifies paragraph-level formatting.

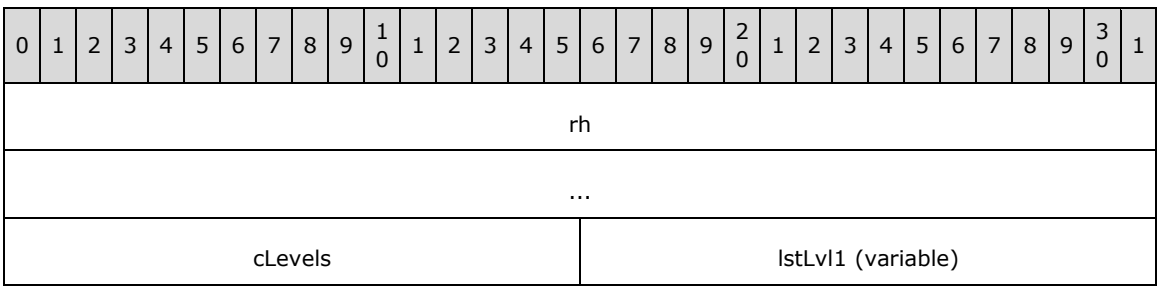
cf9 (variable): A TextCFException2.9.17 structure that specifies character-level formatting.

2.9.39 TextMasterStyle10Atom

Referenced by: PP10DocBinaryTagExtension, PP10SlideBinaryTagExtension

An atom record that specifies additional character-level formatting of a main master slide.

If this **TextMasterStyle10Atom** is contained in a **MainMasterContainer** record (section 2.5.3), character-level formatting not specified by this **TextMasterStyle10Atom** record inherits from the **TextMasterStyle10Atom** records contained in the PP2.4.23.6DocBinaryTagExtension record.



...
lstLvl2 (variable)
...
lstLvl3 (variable)
...
lstLvl4 (variable)
...
lstLvl5 (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	Specifies the type of text to which the formatting applies. It MUST be a **TextTypeEnum** enumeration value.
rh.recType	MUST be an RT_TextMasterStyle10Atom.

cLevels (2 bytes): An unsigned integer that specifies the number of style levels. It MUST be less than or equal to 0x0005.

lstLvl1 (variable): An optional TextMasterStyle2.9.40Level structure that specifies additional master formatting for text that has an IndentLevel equal to 0x0000. It MUST exist if and only if **cLevels** is greater than 0x0000.

lstLvl2 (variable): An optional TextMasterStyle2.9.40Level structure that specifies additional master formatting for text that has an IndentLevel equal to 0x0001. It MUST exist if and only if **cLevels** is greater than 0x0001.

lstLvl3 (variable): An optional TextMasterStyle2.9.40Level structure that specifies additional master formatting for text that has an IndentLevel equal to 0x0002. It MUST exist if and only if **cLevels** is greater than 0x0002.

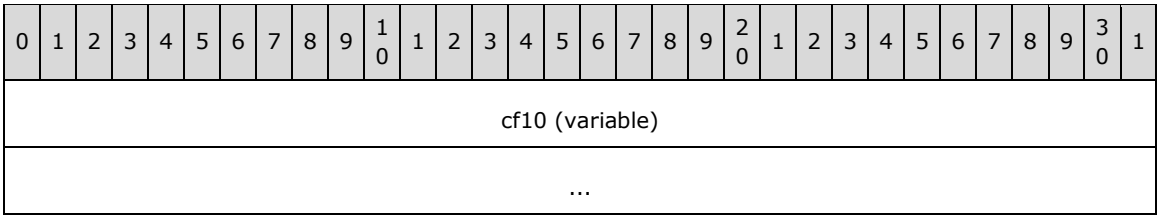
lstLvl4 (variable): An optional TextMasterStyle2.9.40Level structure that specifies additional master formatting for text that has an IndentLevel equal to 0x0003. It MUST exist if and only if **cLevels** is greater than 0x0003.

lstLvl5 (variable): An optional TextMasterStyle2.9.40Level structure that specifies additional master formatting for text that has an IndentLevel equal to 0x0004. It MUST exist if and only if **cLevels** is greater than 0x0004.

2.9.40 TextMasterStyle10Level

Referenced by: TextMasterStyle10Atom

A structure that specifies additional character-level formatting for a style level.



cf10 (variable): A TextCFException2.9.18 structure that specifies additional character-level formatting.

2.9.41 TextHeaderAtom

Referenced by: SlideListWithTextSubContainerOrAtom, TextClientDataSubContainerOrAtom

An atom record that specifies the type of a text body. The presence of this atom indicates a text body whose properties are specified by other atom and container records that follow. The records comprising this text body continue up until the next **SlidePersistAtom** record (section 2.4.14.5) or the next **TextHeaderAtom** record that follows this **TextHeaderAtom** or the end of the **rgChildRec** array of the OfficeArtClientTextBox or **SlideListWithTextContainer** (section 2.4.14.3) record that contains this **TextHeaderAtom**.

Let the *corresponding slide* be specified by one of the following:

- When this **TextHeaderAtom** is contained in a **SlideListWithTextContainer** record, let the *corresponding slide* be the **SlideContainer** record (section 2.5.1) as specified by the **SlidePersistAtom** record that most closely precedes this **TextHeaderAtom** record.
- When this **TextHeaderAtom** is contained in a **SlideContainer** record, let the *corresponding slide* be specified by the **SlideContainer** record that contains this **TextHeaderAtom** record.
- When this **TextHeaderAtom** is contained in a **NotesContainer** record (section 2.5.6), let the *corresponding slide* be specified by the **SlideContainer** record referred to by the **notesAtom.slideIdRef** field of the **NotesContainer** record that contains this **TextHeaderAtom** record.

Let the *corresponding main master* be specified by the **MainMasterContainer** record (section 2.5.3) specified by the **slideAtom.masterIdRef** field of the *corresponding slide.*

Let the *corresponding shape* be specified by one of the following:

- When this **TextHeaderAtom** record is contained in a **OfficeArtSpContainer** record ([MS-ODRAW] section 2.2.14), let the *corresponding shape* be the **OfficeArtSpContainer** record that contains this **TextHeaderAtom** record.
- When this **TextHeaderAtom** record is contained in a **SlideListWithTextContainer** record, let the *corresponding shape* be specified by the **OfficeArtSpContainer** record ([MS-ODRAW] section 2.2.14) contained in the *corresponding slide* that contains an OutlineTextRefAtom record that refers to this **TextHeaderAtom** record.

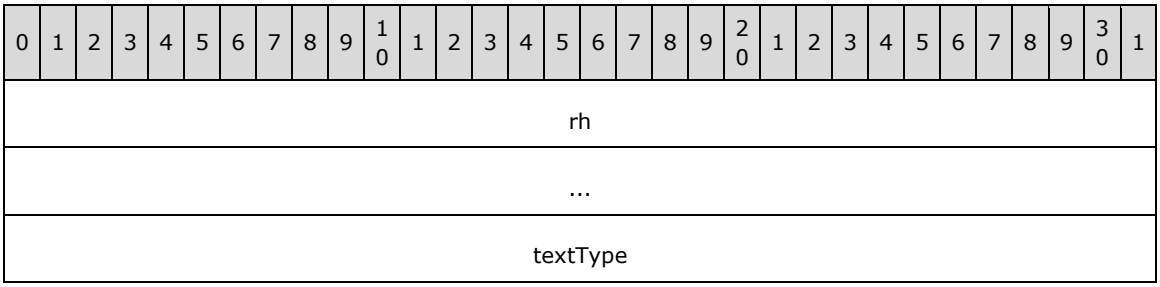
The characters of the text body are specified by the TextCharsAtom record or the TextBytesAtom record, if any, that follows this **TextHeaderAtom** record. In addition, the text body contains a single

terminating paragraph break character (0x000D) that is not included in the TextCharsAtom record or TextBytesAtom record.

Let the *corresponding text style*, if any, be specified by the StyleTextPropAtom record that follows this **TextHeaderAtom**.

The text body contains a sequence of character runs comprised of consecutive characters with identical TextCFException record data as specified by the TextCfRun structures in the **rgTextCfRun** array of the *corresponding text style*.

Let the *corresponding text placeholder list* be as specified in the **SlideListWithTextContainer** record that contains this **TextHeaderAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	Specifies whether this **TextHeaderAtom** record specifies the text of a placeholder shape. It MUST be greater than or equal to 0x000 and less than or equal to 0x005. When this **TextHeaderAtom** record is contained in a **SlideListWithTextContainer** record, this field specifies the index of an item in the *corresponding text placeholder list*. When this **TextHeaderAtom** record is contained in an OfficeArtClientTextbox record, this field MUST be 0x000.
rh.recType	MUST be RT_TextHeaderAtom.
rh.recLen	MUST be 0x00000004.

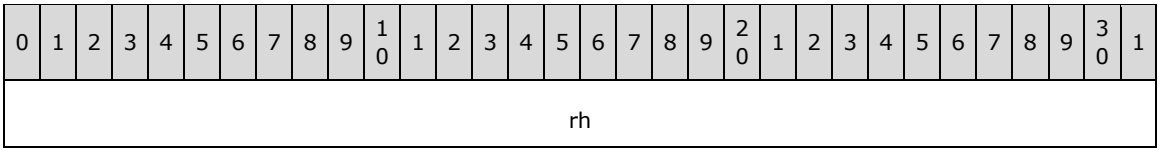
textType (4 bytes): A **TextTypeEnum** enumeration that specifies the type of the text body.

2.9.42 TextCharsAtom

Referenced by: SlideListWithTextSubContainerOrAtom, TextClientDataSubContainerOrAtom

An atom record that specifies Unicode characters.

Let the *corresponding text* be specified by the TextHeaderAtom record that most closely precedes this record.



...
textChars (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TextCharsAtom.
rh.recLen	MUST be even.

textChars (variable): An array of UTF-16 Unicode [RFC2781] characters that specifies the characters of the *corresponding text*. The length, in bytes, of the array is specified by **rh.recLen**. The array MUST NOT contain the NUL character 0x0000.

2.9.43 TextBytesAtom

Referenced by: SlideListWithTextSubContainerOrAtom, TextClientDataSubContainerOrAtom

An atom record that specifies Unicode characters.

Let the *corresponding text* be specified by the TextHeaderAtom record that most closely precedes this record.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
textBytes (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TextBytesAtom.

textBytes (variable): An array of bytes that specifies the characters of the *corresponding text*. Each item represents the low byte of a UTF-16 Unicode [RFC2781] character whose high byte is 0x00.

The length, in bytes, of the array is specified by **rh.recLen**. The array MUST NOT contain a 0x00 byte.

2.9.44 StyleTextPropAtom

Referenced by: SlideListWithTextSubContainerOrAtom, TextClientDataSubContainerOrAtom

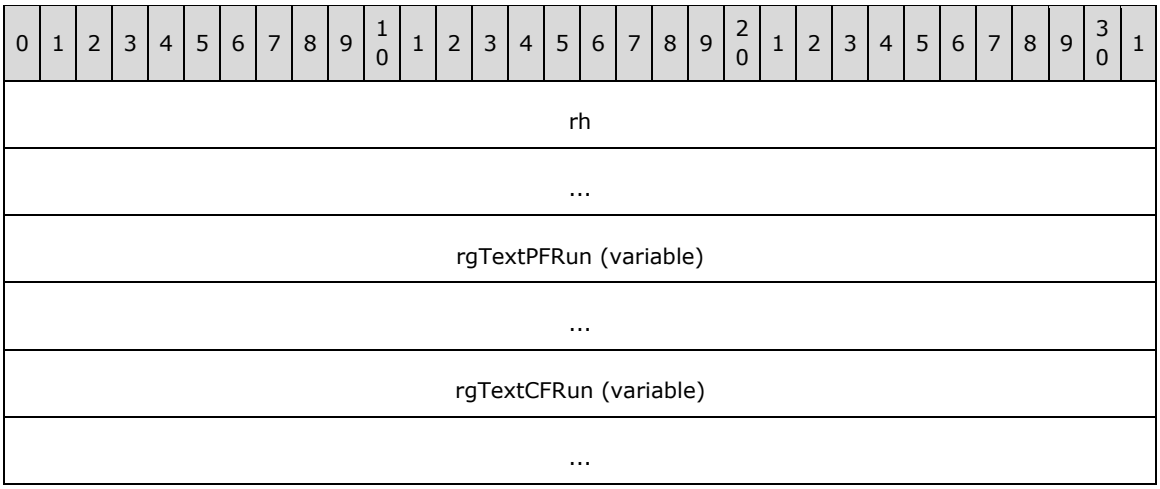
An atom record that specifies character-level and paragraph-level formatting.

Let the *corresponding text* be specified by the TextHeaderAtom record that most closely precedes this record.

Let the *corresponding shape* be as specified in the *corresponding text*.

Let the *corresponding main master* be as specified in the *corresponding text*.

If the *corresponding shape* is a placeholder shape, character-level and paragraph-level formatting not specified by this StyleTextPropAtom record inherit from the TextMasterStyleAtom records contained in the *corresponding main master*.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_StyleTextPropAtom.

rgTextPFRun (variable): An array of TextPFRun structures that specifies paragraph-level formatting for the *corresponding text*. The **count** field of each TextPFRun item specifies the number of characters to which the formatting applies, starting with the character at the zero-based index equal to the sum of the **count** fields of all previous TextPFRun records in the array.

The sum of the **count** fields of the TextPFRun items MUST be equal to the number of characters in the *corresponding text*.

rgTextCFRun (variable): An array of TextCFRun structures that specifies character-level formatting for the *corresponding text*. The **count** field of each TextCFRun specifies the number of characters

to which the formatting applies, starting with the character at the zero-based index equal to the sum of the **count** fields of all previous TextCFRun records in the array.

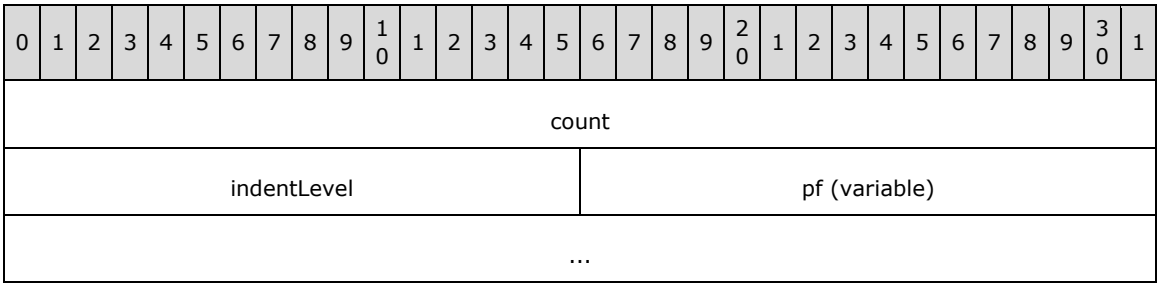
The sum of the **count** fields of the TextCFRun items MUST be equal to the number of characters in the *corresponding text*.

2.9.45 TextPFRun

Referenced by: StyleTextPropAtom

A structure that specifies the paragraph-level formatting of a run of text.

Let the *corresponding text* be as specified in the StyleTextPropAtom record that contains this **TextPFRun** structure.



count (4 bytes): An unsigned integer that specifies the number of characters of the *corresponding text* to which this paragraph formatting applies.

indentLevel (2 bytes): An IndentLevel that specifies the indentation level of the paragraph.

pf (variable): A TextPFException structure that specifies paragraph-level formatting. Sub-fields are further specified in the following table.

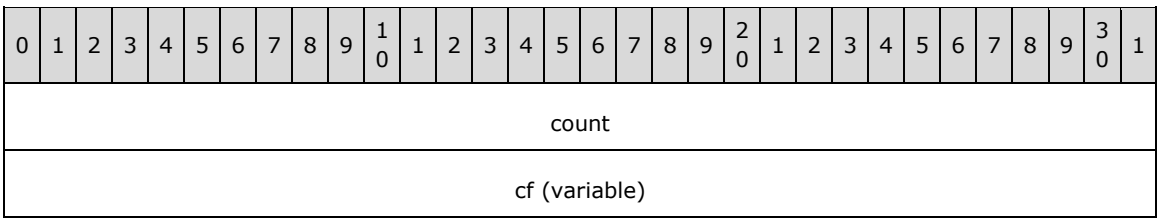
Field	Meaning
masks.leftMargin	MUST be **FALSE**.
masks.indent	MUST be **FALSE**.
masks.defaultTabSize	MUST be **FALSE**.
masks.tabStops	MUST be **FALSE**.

2.9.46 TextCFRun

Referenced by: StyleTextPropAtom

A structure that specifies the character-level formatting of a run of text.

Let the *corresponding text* be as specified in the StyleTextPropAtom record that contains this **TextCFRun** structure.



...

count (4 bytes): An unsigned integer that specifies the number of characters of the corresponding text to which this character formatting applies.

cf (variable): A TextCFException structure that specifies character-level formatting.

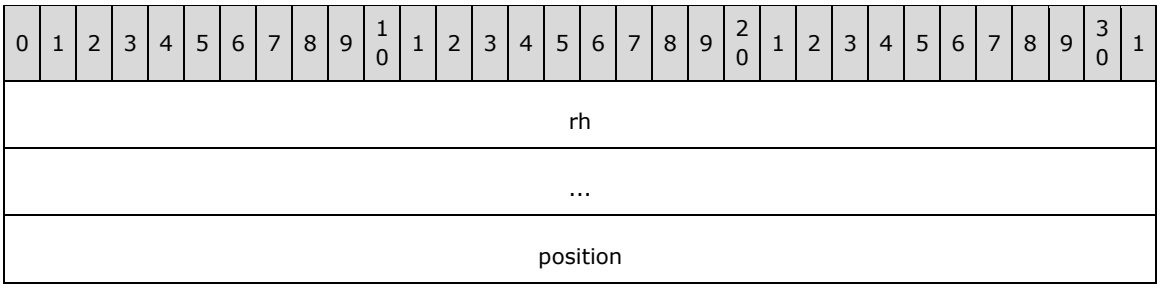
2.9.47 SlideNumberMCAtom

Referenced by: SlideListWithTextSubContainerOrAtom, TextClientDataSubContainerOrAtom

An atom record that specifies a slide number metacharacter.

The metacharacter is replaced by the slide number on which the metacharacter is found.

Let the *corresponding text* be specified by the TextHeaderAtom record that most closely precedes this **SlideNumberMCAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_SlideNumberMetaCharAtom.
rh.recLen	MUST be 0x00000004.

position (4 bytes): A TextPosition that specifies the position of the metacharacter in the *corresponding text*.

2.9.48 HeaderMCAtom

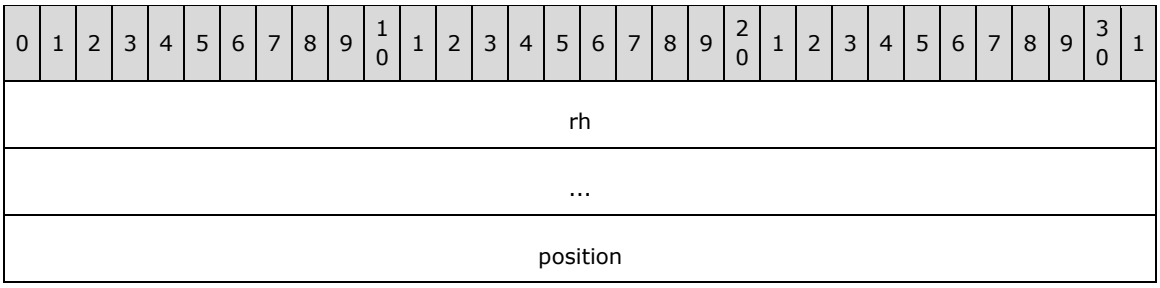
Referenced by: SlideListWithTextSubContainerOrAtom, TextClientDataSubContainerOrAtom

An atom record that specifies a header metacharacter.

The metacharacter is replaced by the text in the *corresponding header*.

Let the *corresponding header* be specified by the HeaderAtom record contained in the **NotesHeadersFootersContainer** record (section 2.4.15.6) for the handout slide or the notes slide.

Let the *corresponding text* be specified by the TextHeaderAtom record that most closely precedes this **HeaderMCAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_HeaderMetaCharAtom.
rh.recLen	MUST be 0x00000004.

position (4 bytes): A TextPosition that specifies the position of the header metacharacter in the *corresponding text*.

2.9.49 FooterMCAtom

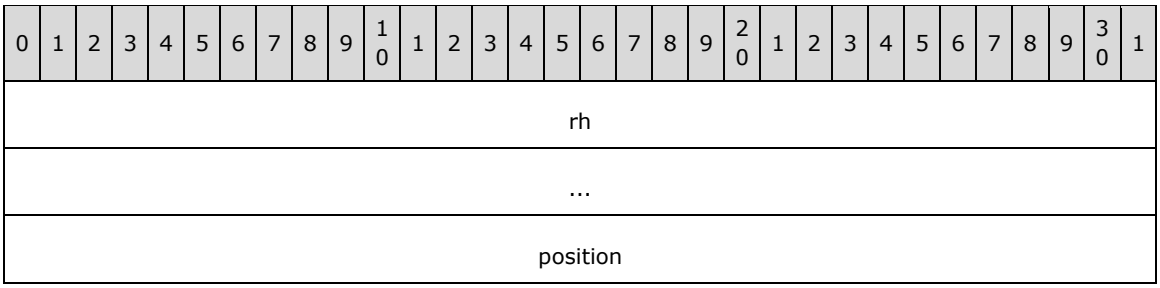
Referenced by: SlideListWithTextSubContainerOrAtom, TextClientDataSubContainerOrAtom

An atom record that specifies a footer metacharacter.

The metacharacter is replaced by the text in the *corresponding footer*.

Let the *corresponding footer* be specified by the FooterAtom record contained in the **SlideHeadersFootersContainer** record (section 2.4.15.1) for presentation slides, the **NotesHeadersFootersContainer** record (section 2.4.15.6) for handout slides and notes slides, or in the PerSlideHeadersFootersContainer record for each slide.

Let the *corresponding text* be specified by the TextHeaderAtom record that most closely precedes this **FooterMCAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.

Field	Meaning
rh.recType	MUST be an RT_FooterMetaCharAtom.
rh.recLen	MUST be 0x00000004.

position (4 bytes): A TextPosition that specifies the position of the metacharacter in the *corresponding text*.

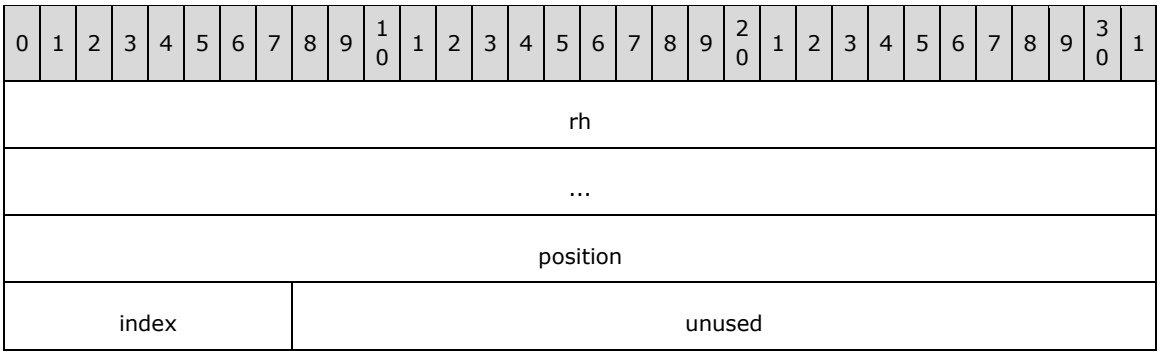
2.9.50 DateTimeMCAtom

Referenced by: SlideListWithTextSubContainerOrAtom, TextClientDataSubContainerOrAtom

An atom record that specifies a datetime metacharacter.

The metacharacter is replaced by the current datetime. Current datetime is formatted in the style that is specified by the **index** field.

Let the corresponding text be specified by the TextHeaderAtom record that most closely precedes this **DateTimeMCAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_DateTimeMetaCharAtom.
rh.recLen	MUST be 0x00000008.

position (4 bytes): A TextPosition that specifies the position of the metacharacter in the *corresponding text*.

index (1 byte): An unsigned byte that specifies the **Format ID** used to stylize datetime. The identifier specified by the **Format ID** is converted based on the LCID [MS-LCID] into a value or string as specified in the following tables. The LCID is specified in **TextSIException.lid**. If no valid LCID is found in **TextSIException.lid, TextSIException.altLid** (if it exists) is used. If, in the following tables, the converted **Format ID** is a value, it specifies the format index (specified in [MS-OSHARED] section 2.4.4.1) that is used to style the datetime. If, in the following tables, the converted Format ID is a string, that string is used as the style to format the datetime. The value MUST be greater than or equal to 0x0 and MUST be less than or equal to 0xC.

LCID: American (0x0409)

Format ID	Converted value/string
0	0
1	1
2	8
3	2
4	5
5	9
6	10
7	11
8	12
9	15
10	16
11	13
12	14

LCID: British (0x0809) / Australian (0x0C09)

Format ID	Converted value/string
0	0
1	1
2	"d MMMM, yyy"
3	2
4	5
5	9
6	10
7	11
8	12
9	15
10	16
11	13
12	14

LCID: Japanese (0x0411)

Format ID	Converted value/string
0	4
1	8
2	7
3	3
4	0
5	9
6	5
7	11
8	12
9	"HH:mm"
10	"HH:mm:ss"
11	15
12	16

LCID: Taiwan (0x0404)

Format ID	Converted value/string
0	0
1	1
2	3
3	7
4	12
5	9
6	10
7	4
8	11
9	"HH:mm"
10	"HH:mm:ss"
11	"H:mm AMPM"
12	"H:mm:ss AMPM"

LCID: China (0x0804)

Format ID	Converted value/string
0	0
1	1
2	2
3	2
4	4
5	9
6	5
7	"\x79\x79\x79\x79\x5E74\x4D\x6708\x64\x65E5\x68\x65F6\x6D\x5206"
8	"\x79\x79\x79\x79\x5E74\x4D\x6708\x64\x65E5\x661F \x671f\x57\x68\x65F6\x6D\x5206\x73\x79D2"
9	"HH:mm"
10	"HH:mm:ss"
11	"\x41\x4D\x50\x4D\x68\x65F6\x6D\x5206"
12	"\x41\x4D\x50\x4D\x68\x65F6\x6D\x5206\x73\x79D2"

LCID: Korean (0x0412)

Format ID	Converted value/string
0	0
1	1
2	6
3	3
4	4
5	10
6	7
7	12
8	11
9	"HH:mm"
10	"HH:mm:ss"
11	13
12	14

LCID: Arabic (0x0401)

Format ID	Converted value/string
0	0
1	1
2	2
3	3
4	4
5	5
6	8
7	7
8	8
9	1
10	10
11	11
12	5

LCID: Hebrew (0x040D)

Format ID	Converted value/string
0	0
1	1
2	2
3	6
4	11
5	5
6	12
7	7
8	8
9	9
10	1
11	11
12	6

LCID: Swedish (0x041D)

Format ID	Converted value/string
0	0
1	1
2	3
3	2
4	7
5	9
6	10
7	11
8	12
9	15
10	16
11	13
12	14

LCID: Singapore (0x1004) / Macao SAR (0x1404) / Hong Kong SAR (0x0C04)

Format ID	Converted value/string
0	0
1	1
2	3
3	2
4	4
5	9
6	5
7	"\x79\x79\x79\x79\x5E74\x4D\x6708\x64\x65E5\x68\x65F6\x6D\x5206"
8	"\x79\x79\x79\x79\x5E74\x4D\x6708\x64\x65E5\x661F \x671f\x57\x68\x65F6\x6D\x5206\x73\x79D2"
9	"HH:mm"
10	"HH:mm:ss"
11	"\x41\x4D\x50\x4D\x68\x65F6\x6D\x5206"
12	"\x41\x4D\x50\x4D\x68\x65F6\x6D\x5206\x73\x79D2"

LCID: Thai (0x41E)

Format ID	Converted value/string
0	0
1	1
2	2
3	3
4	5
5	6
6	7
7	8
8	9
9	10
10	11
11	13
12	14

LCID: Vietnamese (0x042A)

Format ID	Converted value/string
0	0
1	1
2	2
3	3
4	5
5	6
6	10
7	11
8	12
9	13
10	14
11	15
12	16

LCID: Hindi (0x0439)

Format ID	Converted value/string
0	1
1	2
2	3
3	5
4	7
5	11
6	13
7	0
8	1
9	5
10	10
11	11
12	14

LCID: Syriac (0x045A)

Format ID	Converted value/string
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
2	12

unused (3 bytes): Undefined and MUST be ignored.

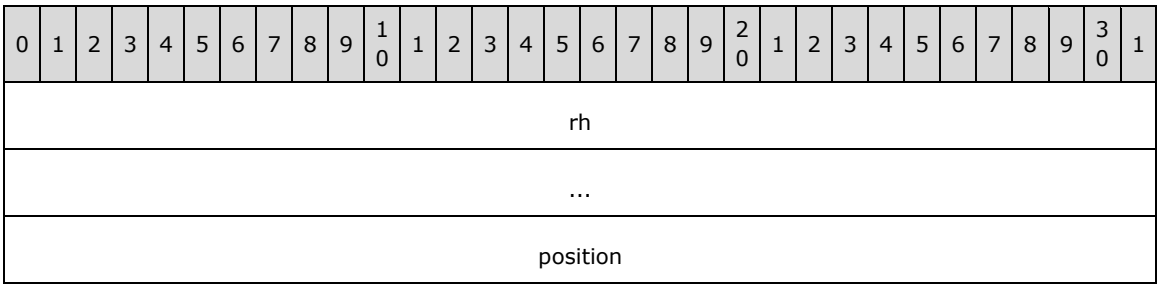
2.9.51 GenericDateMCAtom

Referenced by: SlideListWithTextSubContainerOrAtom, TextClientDataSubContainerOrAtom

An atom record that specifies a datetime metacharacter.

The metacharacter is replaced by the datetime option specified in the HeadersFootersAtom record.

Let the *corresponding text* be specified by the TextHeaderAtom record that most closely precedes this **GenericDateMCAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_GenericDateMetaCharAtom.
rh.recLen	MUST be 0x00000004

position (4 bytes): A TextPosition that specifies the position of the metacharacter in the *corresponding text*.

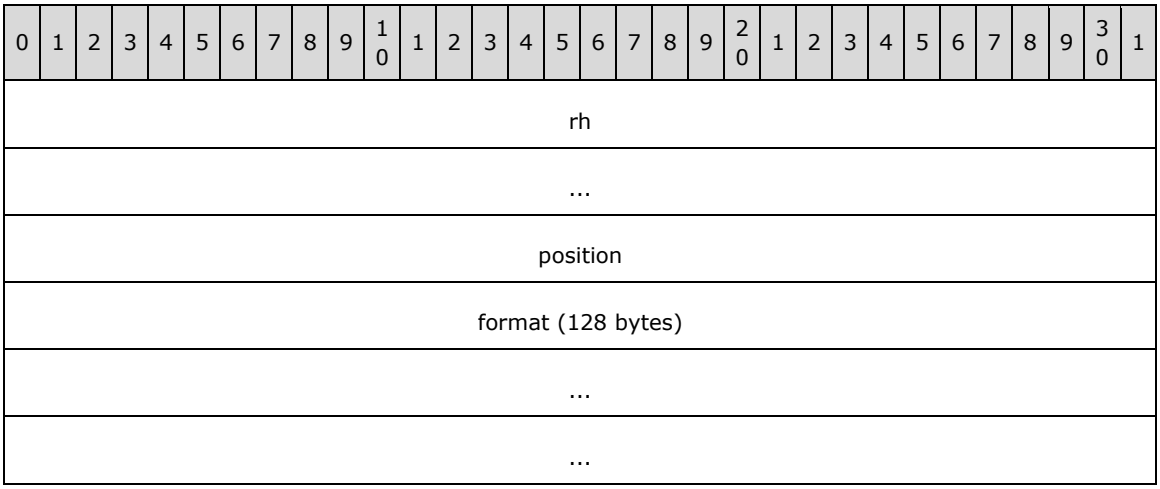
2.9.52 RTFDateTimeMCAtom

Referenced by: SlideListWithTextSubContainerOrAtom, TextClientDataSubContainerOrAtom

An atom record that specifies a Rich Text Format (RTF) datetime metacharacter. RTF format is specified by [MSFT-RTF].

The metacharacter is replaced by the datetime, using the format specified in the format string in this metacharacter.

Let the *corresponding text* be specified by the TextHeaderAtom record that most closely precedes this **RTFDateTimeMCAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_RtfDateTimeMetaCharAtom.
rh.recLen	MUST be 0x00000084.

position (4 bytes): A TextPosition that specifies the position of the metacharacter in the *corresponding text*.

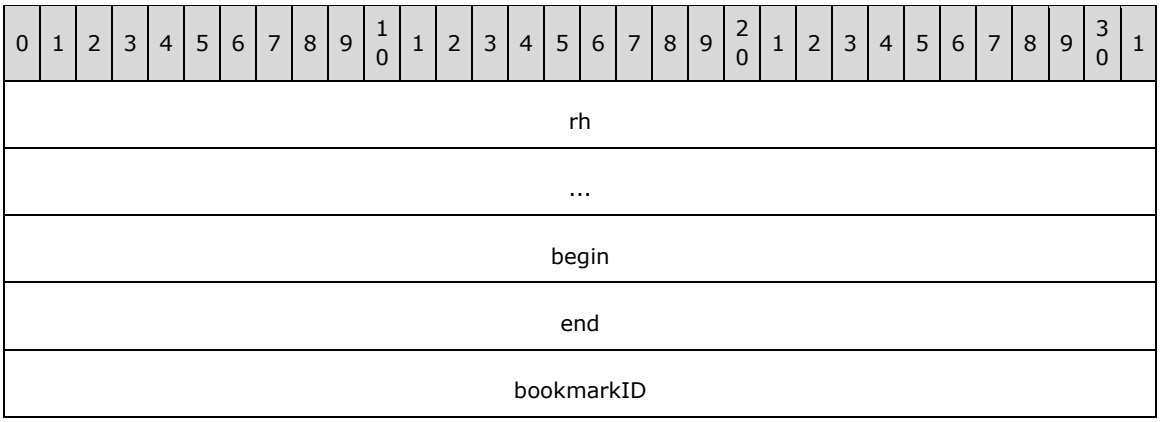
format (128 bytes): A char2.2.4 that specifies the date and time format in RTF.

2.9.53 TextBookmarkAtom

Referenced by: SlideListWithTextSubContainerOrAtom, TextClientDataSubContainerOrAtom

An atom record that specifies a range of text that has a bookmark. The length of the range of text is specified by the following formula:

```
end - begin
```



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TextBookmarkAtom.
rh.recLen	MUST be 0x0000000C.

begin (4 bytes): A TextPosition that specifies the beginning of the bookmarked range.

end (4 bytes): A TextPosition that specifies the end of the bookmarked range. This field MUST be greater than **begin** and SHOULD<107> be less than or equal to **begin** plus 255.

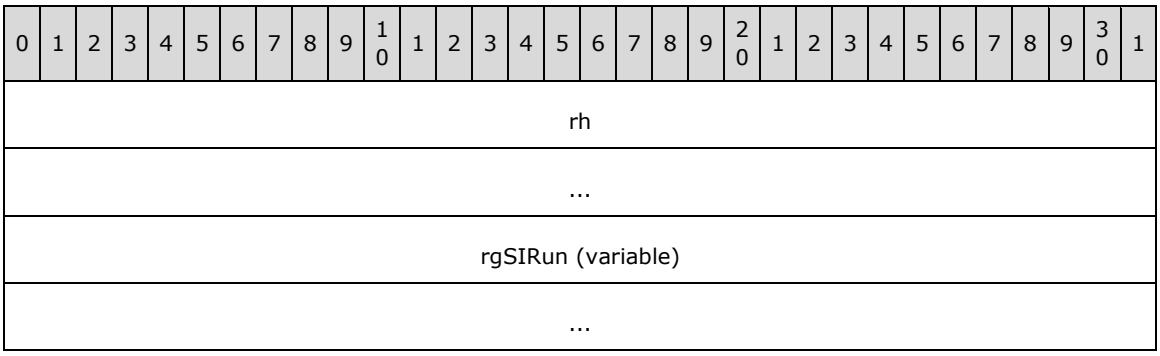
bookmarkID (4 bytes): An unsigned integer that specifies a reference to a bookmark identifier. It MUST be the same as the **bookmarkID** field of a BookmarkEntityAtom record.

2.9.54 TextSpecialInfoAtom

Referenced by: SlideListWithTextSubContainerOrAtom, TextClientDataSubContainerOrAtom

An atom record that specifies additional text properties.

Let the *corresponding text* be specified by the TextHeaderAtom record that most closely precedes this record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TextSpecialInfoAtom.

rgSIRun (variable): An array of TextSIRun structures that specifies additional text properties for the *corresponding text*. The **count** field of each TextSIRun specifies the number of characters to which the properties apply, starting with the character at the zero-based index equal to the sum of the **count** fields of all previous TextSIRun records in the array.

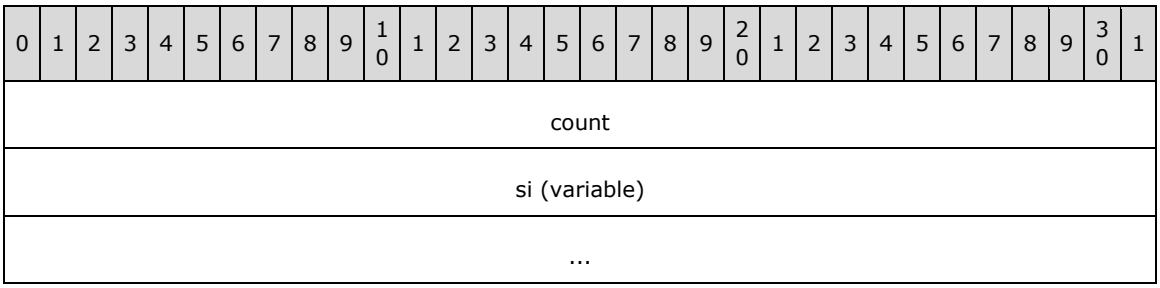
The sum of the **count** fields of the TextSIRun items MUST be equal to the number of characters in the *corresponding text*. The length, in bytes, of the array is specified by **rh.recLen**.

2.9.55 TextSIRun

Referenced by: TextSpecialInfoAtom

A structure that specifies language and spelling information for a run of text.

Let the *corresponding text* be as specified in the TextSpecialInfoAtom record that contains this **TextSIRun** structure.



count (4 bytes): An **unsigned integer** that specifies the number of characters of the corresponding text to which these additional text properties apply. It MUST be greater than or equal to 0x00000001.

si (variable): A TextSIException structure that specifies language and spelling information.

2.9.56 TextInteractiveInfoInstance

Referenced by: SlideListWithTextSubContainerOrAtom, TextClientDataSubContainerOrAtom

A variable type record whose type and meaning are dictated by the value of **rh.recInstance**, as specified in the following table.

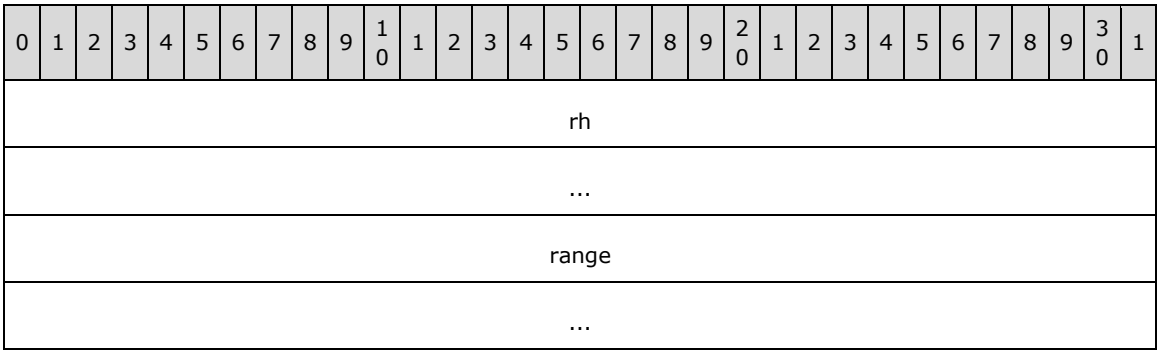
Value	Meaning
0x000	A MouseClickTextInteractiveInfoAtom record that specifies the corresponding text range of the preceding MouseClickInteractiveInfoContainer record.
0x001	A MouseOverTextInteractiveInfoAtom record that specifies the corresponding text range of the preceding MouseOverInteractiveInfoContainer record.

2.9.57 MouseClickTextInteractiveInfoAtom

Referenced by: TextInteractiveInfoInstance

An atom record that specifies a text range that anchors the preceding MouseClickInteractiveInfoContainer record in the containing OfficeArtClientTextbox record or **SlideListWithTextContainer** record (section 2.4.14.3).

Let the *corresponding text* be specified by the TextHeaderAtom record that most closely precedes this **MouseClickTextInteractiveInfoAtom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TextInteractiveInfoAtom.
rh.recLen	MUST be 0x00000008.

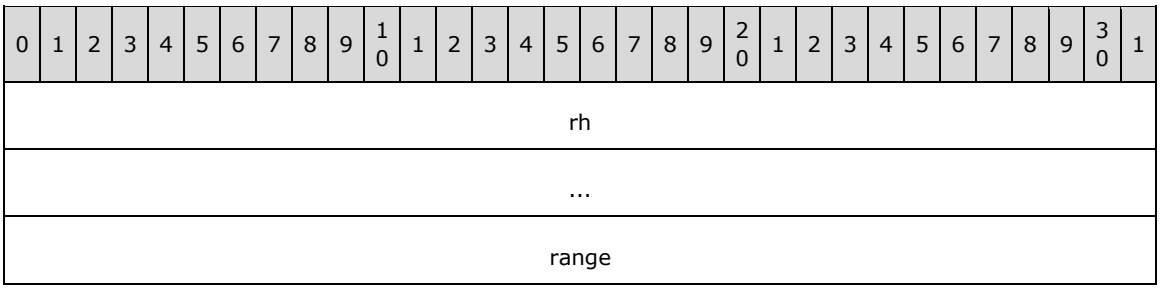
range (8 bytes): A TextRange structure that specifies the anchor in the *corresponding text*.

2.9.58 MouseOverTextInteractiveInfoAtom

Referenced by: TextInteractiveInfoInstance

An atom record that specifies a text range that anchors the preceding MouseOverInteractiveInfoContainer record in the containing OfficeArtClientTextbox record or **SlideListWithTextContainer** record (section 2.4.14.3).

Let the *corresponding text* be specified by the TextHeaderAtom record that most closely precedes this **MouseOverTextInteractiveInfoAtom** record.



...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be an RT_TextInteractiveInfoAtom.
rh.recLen	MUST be 0x00000008.

range (8 bytes): A TextRange structure that specifies the anchor in the *corresponding text*.

2.9.59 TextRange

Referenced by: MouseClickTextInteractiveInfoAtom, MouseOverTextInteractiveInfoAtom

A structure that specifies a range of text.

Let the *corresponding text* be as specified in the MouseClickTextInteractiveInfoAtom record or the MouseOverTextInteractiveInfoAtom record that contains this **TextRange** structure.

The range specified must be valid for the *corresponding text*. The length of the range of text is specified by the following formula:

```
end - begin
```

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
begin																															
end																															

begin (4 bytes): A TextPosition that specifies the first position of the range. It MUST be greater than or equal to zero and MUST be less than or equal to the length of the *corresponding text*.

end (4 bytes): A TextPosition that specifies the cutoff position for the range. The character before this position is the last character in the range. It MUST be greater than **begin** and MUST be less than or equal to the length of the *corresponding text*.

2.9.60 OutlineTextProps9Container

Referenced by: PP9DocBinaryTagExtension

A container record that specifies additional text properties for outline text.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															

...
rgOutlineTextProps9Entry (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_OutlineTextProps9.

rgOutlineTextProps9Entry (variable): An array of OutlineTextProps2.9.61Entry structures that specifies the text properties. The size, in bytes, of the array is specified by **rh.recLen**.

2.9.61 OutlineTextProps9Entry

Referenced by: OutlineTextProps9Container

A structure that specifies additional text properties for a single placeholder shape position on a slide.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
outlineTextHeaderAtom (16 bytes)																															
...																															
...																															
styleTextProp9Atom (variable)																															
...																															

outlineTextHeaderAtom (16 bytes): An OutlineTextPropsHeaderExAtom record that specifies to which placeholder shape position and slide the **styleTextProp9Atom** field applies.

styleTextProp9Atom (variable): A StyleTextProp2.9.67Atom record that specifies additional text properties.

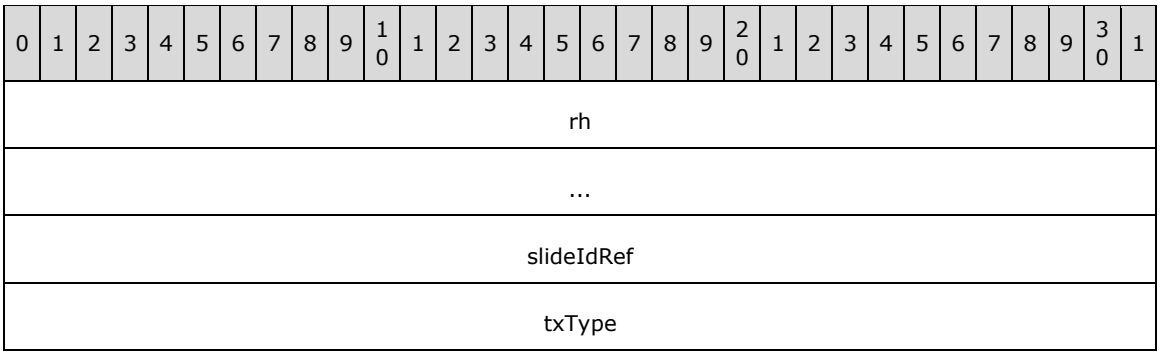
2.9.62 OutlineTextPropsHeaderExAtom

Referenced by: OutlineTextProps10Entry, OutlineTextProps11Entry, OutlineTextProps9Entry

An atom record that specifies a reference to text contained in the **SlideListWithTextContainer** record (section 2.4.14.3).

Let the *corresponding slide persist* be specified by the **SlidePersistAtom** record (section 2.4.14.5) contained in the **SlideListWithTextContainer** record whose **slideId** field is equal to **slideIdRef**.

Let the *corresponding text* be specified by the TextHeaderAtom record referenced by **rh.recInstance**.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	Specifies a zero-based index into the sequence of TextHeaderAtom records that follows the *corresponding slide persist*. It MUST be greater than or equal to 0x000 and less than the number of TextHeaderAtom records that follow the *corresponding slide persist*. It MUST be less than or equal to 0x005.
rh.recType	MUST be RT_OutlineTextPropsHeader9Atom.

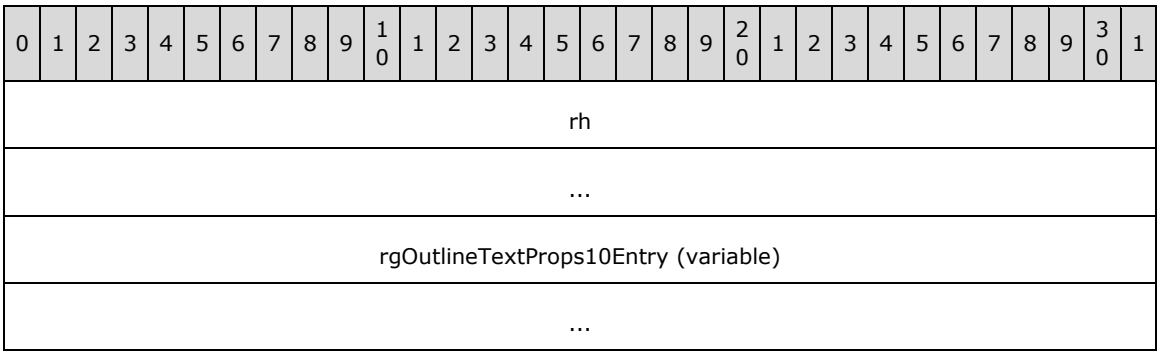
slideIdRef (4 bytes): A **SlideIdRef** (section 2.2.25) that specifies the presentation slide that contains the *corresponding text*. If this field does not reference a valid presentation slide, the structure that contains this **OutlineTextPropsHeaderExAtom** MUST be ignored.

txType (4 bytes): A **TextTypeEnum** enumeration that specifies the type of text of the *corresponding text*.

2.9.63 OutlineTextProps10Container

Referenced by: PP10DocBinaryTagExtension

A container record that specifies additional text properties for outline text.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

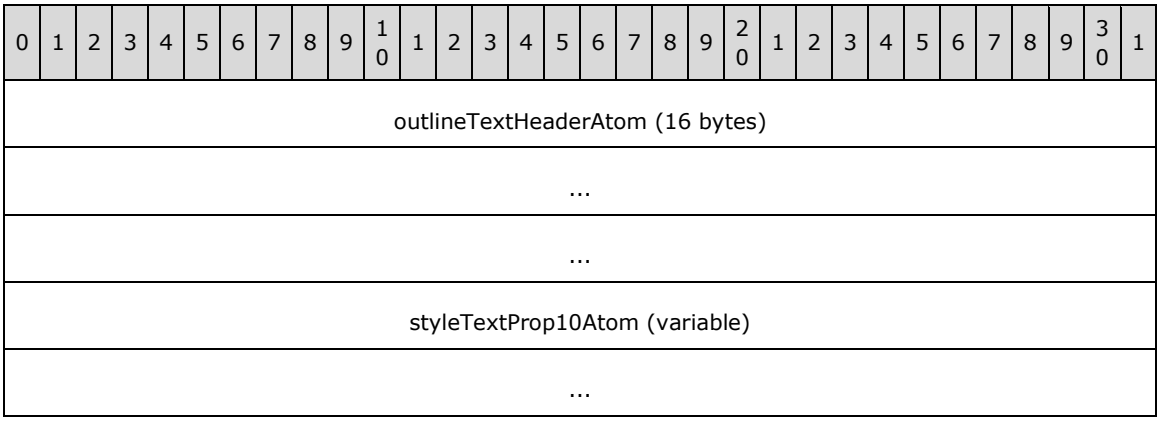
Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_OutlineTextProps10.

rgOutlineTextProps10Entry (variable): An array of OutlineTextProps2.9.64Entry structures that specifies the text properties. The size, in bytes, of the array is specified by **rh.recLen**.

2.9.64 OutlineTextProps10Entry

Referenced by: OutlineTextProps10Container

A structure that specifies additional text properties for a single placeholder shape position on a slide.



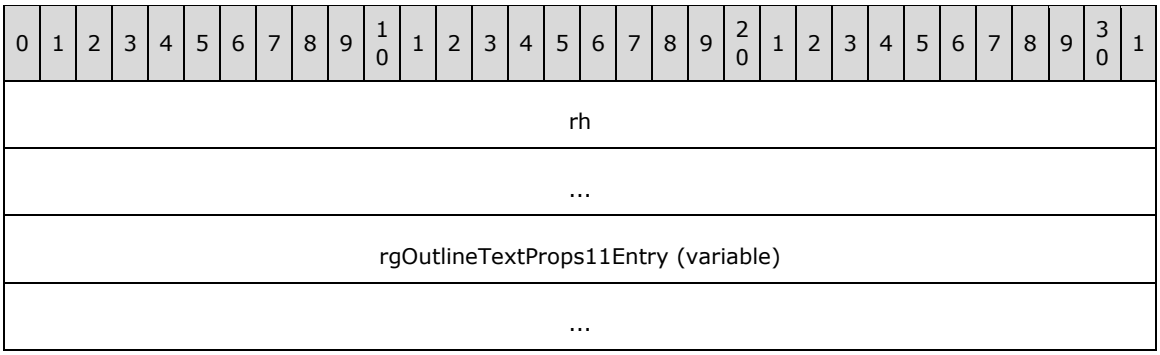
outlineTextHeaderAtom (16 bytes): An OutlineTextPropsHeaderExAtom record that specifies to which placeholder shape position and slide the **styleTextProp10Atom** field applies.

styleTextProp10Atom (variable): A StyleTextProp2.9.69Atom record that specifies additional text properties.

2.9.65 OutlineTextProps11Container

Referenced by: PP11DocBinaryTagExtension

A container record that specifies additional text properties for outline text.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

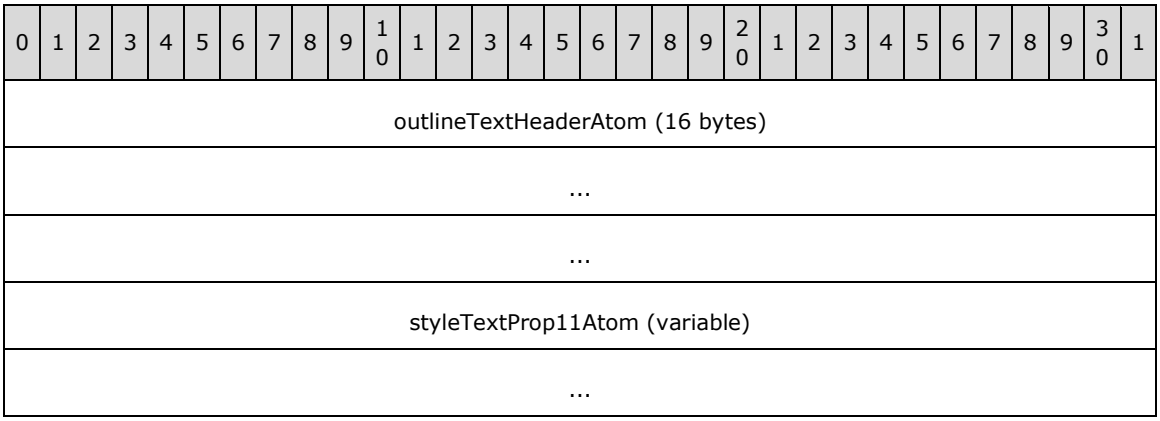
Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_OutlineTextProps11.

rgOutlineTextProps11Entry (variable): An array of OutlineTextProps2.9.66Entry structures that specifies the text properties. The size, in bytes, of the array is specified by **rh.recLen**.

2.9.66 OutlineTextProps11Entry

Referenced by: OutlineTextProps11Container

A structure that specifies additional text properties for a single placeholder shape position on a slide.



outlineTextHeaderAtom (16 bytes): An OutlineTextPropsHeaderExAtom record that specifies to which placeholder shape position and slide the **styleTextProp11Atom** field applies

styleTextProp11Atom (variable): A StyleTextProp2.9.70Atom record that specifies additional text properties.

2.9.67 StyleTextProp9Atom

Referenced by: OutlineTextProps9Entry, PP9ShapeBinaryTagExtension

An atom record that specifies additional text formatting.

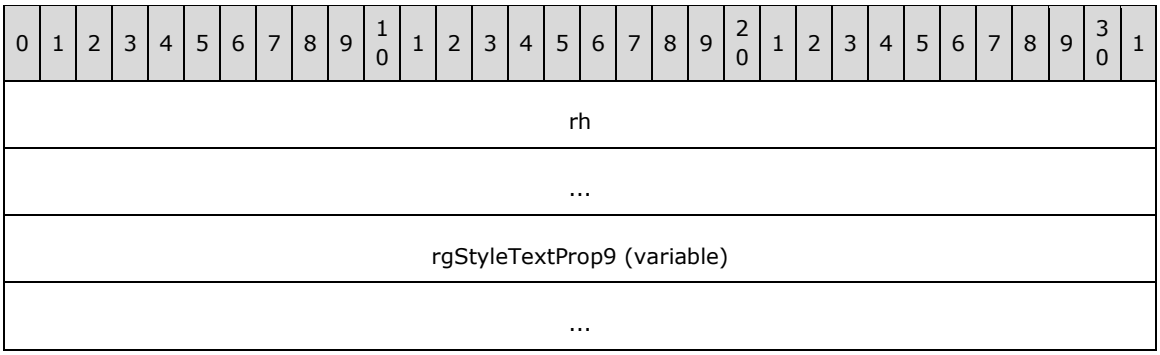
When this record is contained in an OutlineTextProps2.9.61Entry structure, let the *corresponding text* be as specified in the OutlineTextPropsHeaderExAtom record contained in the OutlineTextProps2.9.61Entry structure that contains this **StyleTextProp9Atom** record.

When this record is contained in a PP2.7.18ShapeBinaryTagExtension record, let the *corresponding text* be specified by the TextHeaderAtom record contained in the **OfficeArtSpContainer** ([MS-ODRAW] section 2.2.14) that contains this **StyleTextProp9Atom** record.

Let the *corresponding shape* be as specified in the *corresponding text*.

Let the *corresponding main master* be as specified in the *corresponding text*.

If the *corresponding shape* is a placeholder shape, character-level and paragraph-level formatting not specified by this **StyleTextProp9Atom** record inherit from the TextMasterStyle2.9.37Atom records contained in the *corresponding main master*.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

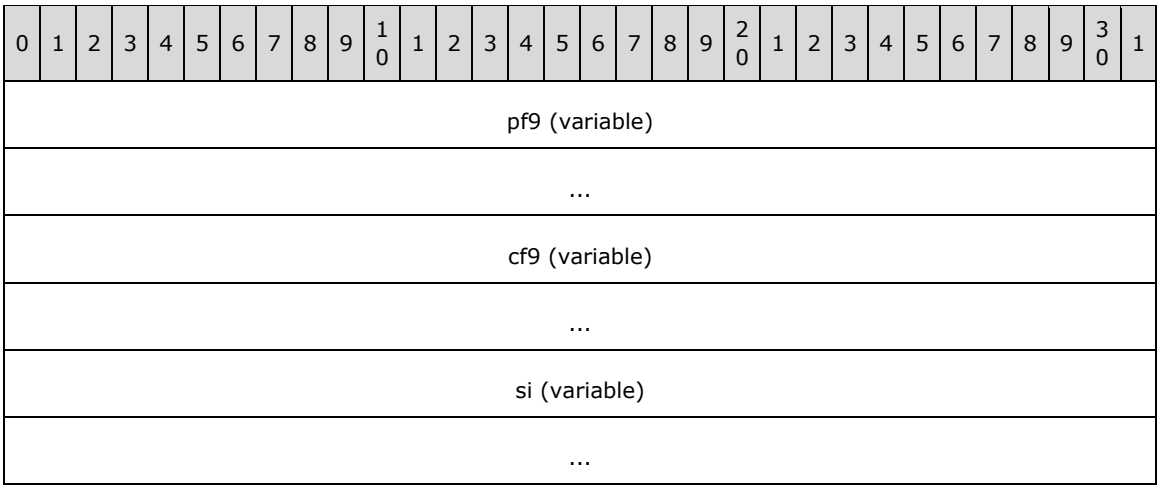
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_StyleTextProp9Atom.

rgStyleTextProp9 (variable): An array of StyleTextProp2.9.68 structures that specifies additional formatting for the *corresponding text*. Each item in the array specifies formatting for a sequence of consecutive character runs of the *corresponding text* that share the same value of the **fontStyle.pp9rt** field of the TextCFException record. If a TextCFException record does not specify a **fontStyle.pp9rt** field, its value is assumed to be 0x0000. An item at index *i* MUST be ignored if `i % 16` is not equal to the value of the **fontstyle.pp9rt** field of the next such sequence. The length, in bytes, of the array is specified by **rh.recLen**.

2.9.68 StyleTextProp9

Referenced by: StyleTextProp9Atom

A structure that specifies additional paragraph-level formatting, character-level formatting, and text properties for a text run.



pf9 (variable): A TextPFException2.9.26 structure that specifies additional paragraph-level formatting.

cf9 (variable): A TextCFException2.9.17 structure that specifies additional character-level formatting.

si (variable): A TextSIException structure that specifies additional text properties. Sub-fields are further specified in the following table.

Field	Meaning
masks.spell	MUST be 0.
masks.lang	MUST be 0.
masks.altLang	MUST be 0.
masks.smartTag	MUST be 0.

2.9.69 StyleTextProp10Atom

Referenced by: OutlineTextProps10Entry, PP10ShapeBinaryTagExtension

An atom record that specifies additional character-level formatting.

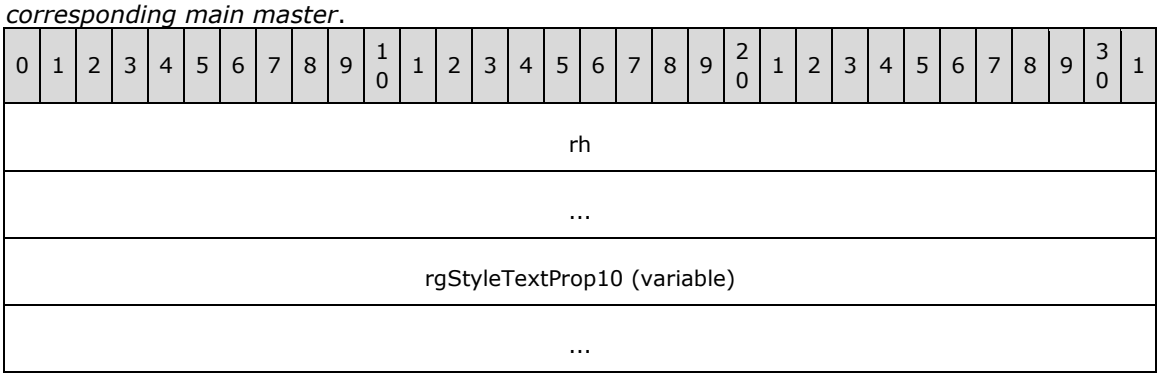
When this record is contained in an OutlineTextProps2.9.64Entry structure, let the *corresponding text* be as specified in the OutlineTextPropsHeaderExAtom record contained in the OutlineTextProps2.9.64Entry structure that contains this **StyleTextProp10Atom** record.

When this record is contained in a PP2.7.19ShapeBinaryTagExtension record, let the *corresponding text* be specified by the TextHeaderAtom record contained in the OfficeArtSpContainer ([MS-ODRAW] section 2.2.14) that contains this **StyleTextProp10Atom** record.

Let the *corresponding shape* be as specified in the *corresponding text*.

Let the *corresponding main master* be as specified in the *corresponding text*.

If the *corresponding shape* is a placeholder shape, character-level formatting not specified by this **StyleTextProp10Atom** record inherits from the TextMasterStyle2.9.39Atom records contained in the *corresponding main master*.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_StyleTextProp10Atom.

rgStyleTextProp10 (variable): An array of TextCFException2.9.18 structures that specifies additional character-level formatting for the *corresponding text*. Each item in the array specifies formatting for a sequence of consecutive character runs that share the same value of the **pp10runid** field of the TextCFException2.9.17 structure, as specified by the StyleTextProp2.9.67Atom record that refers to the same *corresponding text*. If a TextCFException2.9.17 structure does not contain the **pp10runid** field, its value is assumed to be 0x0000. An item at index *i* MUST be ignored if `i % 16` is not equal to the value of the **pp10runid** field of the next such sequence. The length, in bytes, of the array is specified by **rh.recLen**.

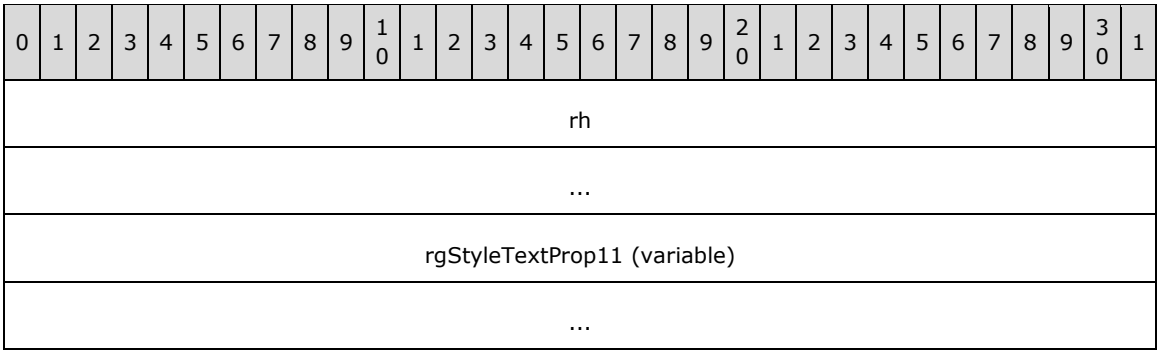
2.9.70 StyleTextProp11Atom

Referenced by: OutlineTextProps11Entry, PP11ShapeBinaryTagExtension

An atom record that specifies additional text properties.

When this record is contained in an OutlineTextProps2.9.66Entry structure, let the *corresponding text* be as specified in the OutlineTextPropsHeaderExAtom record contained in the OutlineTextProps2.9.66Entry structure that contains this **StyleTextProp11Atom** record.

When this record is contained in a PP2.7.20ShapeBinaryTagExtension record, let the *corresponding text* be specified by the TextHeaderAtom record contained in the OfficeArtSpContainer ([MS-ODRAW] section 2.2.14) that contains this **StyleTextProp11Atom** record.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

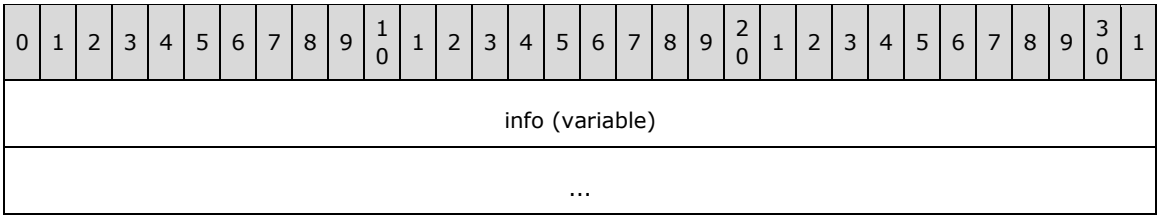
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_StyleTextProp11Atom.

rgStyleTextProp11 (variable): An array of StyleTextProp2.9.71 structures that specifies smart tags for the *corresponding text*. Each item specifies properties for a sequence of characters until the beginning or end of the next smart tag. An item at index *i* MUST be ignored if `i % 16` is not equal to the value of the **si.pp10runid** field of the StyleTextProp2.9.68 record that refers to the same characters. If the StyleTextProp2.9.68 record does not contain a **si.pp10runid** field, its value is assumed to be 0x0000.

2.9.71 StyleTextProp11

Referenced by: StyleTextProp11Atom

A structure that specifies additional text properties.



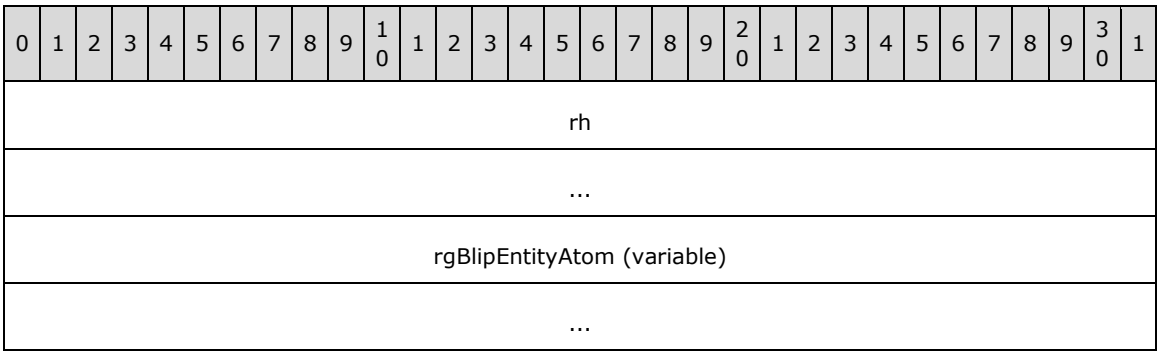
info (variable): A TextSIException structure that specifies additional text properties for a text run. Sub-fields are further specified in the following table.

Field	Meaning
info.spell	MUST be zero.
info.lang	MUST be zero.
info.altLang	MUST be zero.
info.fPp10ext	MUST be zero.
info.fBidi	MUST be zero.

2.9.72 BlipCollection9Container

Referenced by: PP9DocBinaryTagExtension

A container record that specifies information about picture bullet points.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

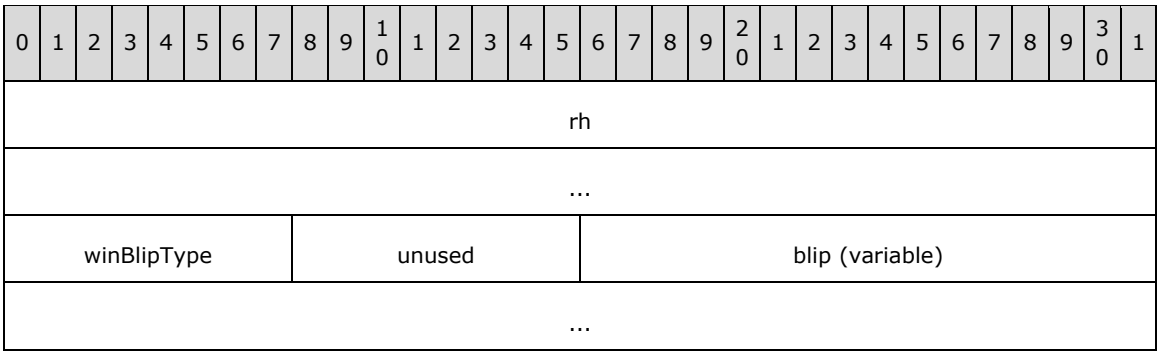
Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_BlipCollection9.

rgBlipEntityAtom (variable): An array of BlipEntityAtom record that specifies picture bullets. It MUST NOT contain more than one BlipEntityAtom record with the same value of **rh.recInstance**. The size, in bytes, of the array is specified by **rh.recLen**.

2.9.73 BlipEntityAtom

Referenced by: BlipCollection9Container

An atom record that specifies a picture bullet.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	An **unsigned integer** that specifies a zero-based index of a picture bullet within the collection of picture bullets specified by **BlipCollection9Container** (section 2.9.72). It MUST be greater than or equal to 0x000 and less than or equal to 0x080.
rh.recType	MUST be an RT_BlipEntity9Atom.

winBlipType (1 byte): An **unsigned integer** that specifies the preferred picture type. It MUST be one of the values in the following table.

Value	Meaning
0x02	Windows Enhanced Metafile [MS-EMF]
0x03	Windows Metafile [MS-WMF]
0x05	JPEG [JFIF]
0x06	PNG [RFC2083]

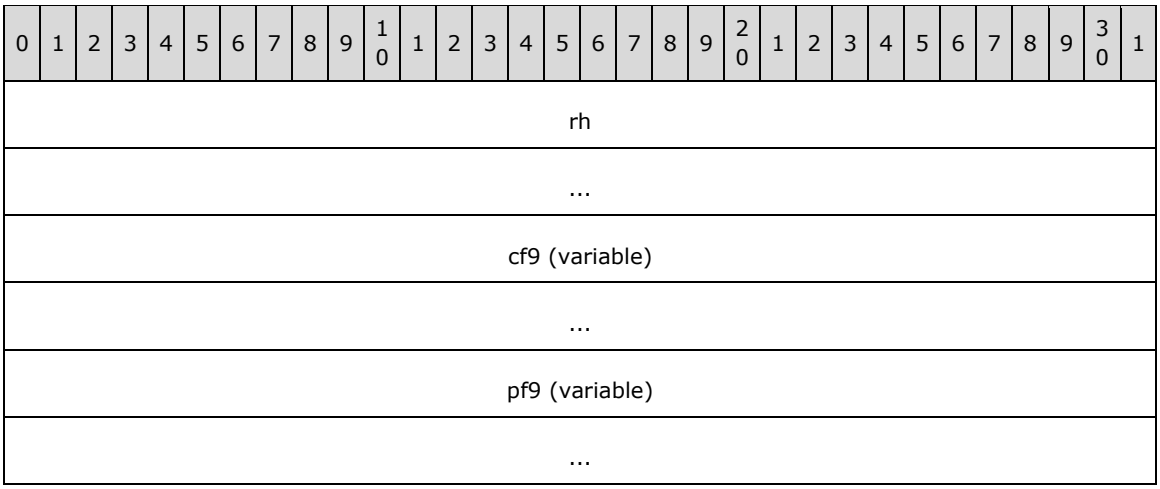
unused (1 byte): Undefined and MUST be ignored.

blip (variable): An **OfficeArtBStoreContainerFileBlock** ([MS-ODRAW] section 2.2.22) that specifies the picture data for the picture bullet.

2.9.74 TextDefaults9Atom

Referenced by: PP9DocBinaryTagExtension

An atom record that specifies additional default character-level and paragraph-level formatting.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TextDefaults9Atom.

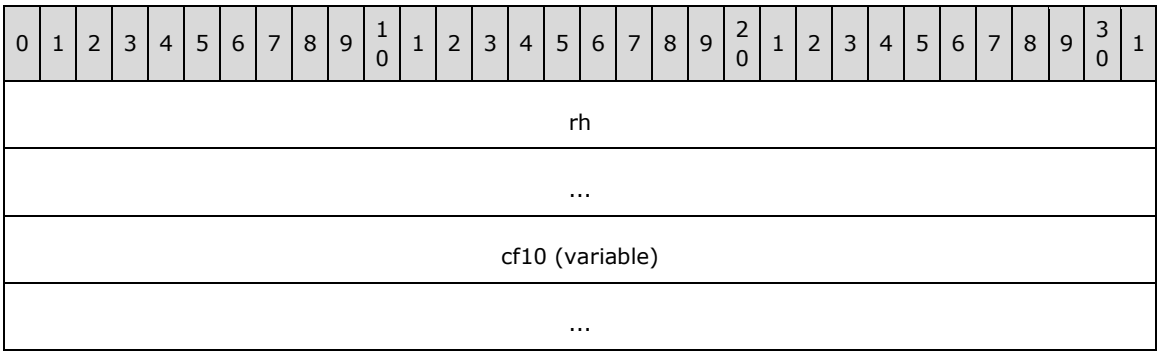
cf9 (variable): A TextCFException2.9.17 structure that specifies default character-level formatting.

pf9 (variable): A TextPFException2.9.26 structure that specifies default paragraph-level formatting.

2.9.75 TextDefaults10Atom

Referenced by: PP10DocBinaryTagExtension

An atom record that specifies additional default character-level formatting.



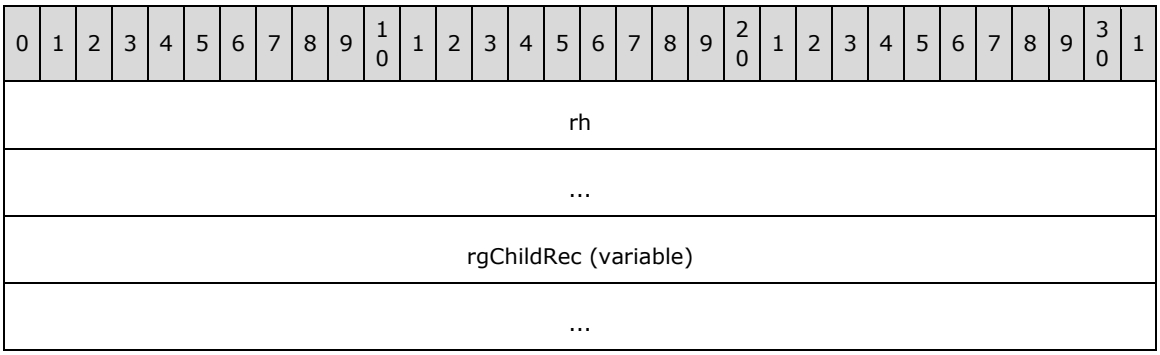
rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_TextDefaults10Atom.

cf10 (variable): A TextCFException2.9.18 structure that specifies additional font information.

2.9.76 OfficeArtClientTextbox

A container record that specifies text related data for a shape.



rh (8 bytes): An **OfficeArtRecordHeader** ([MS-ODRAW] section 2.2.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be 0xF00D.

rgChildRec (variable): An array of TextClientDataSubContainerOrAtom records that specifies text-related data. The size, in bytes, of the array is specified by **rh.recLen**. The sequence of the **rh.recType** fields of the array items MUST be a valid OfficeArtClientTextBoxAtoms as specified in the following ABNF (specified in [RFC5234]) grammar:

```
OfficeArtClientTextBoxAtoms = OutlineAtom / (RT_TextHeaderAtom [RT_TextCharsAtom /
RT_TextBytesAtom]TextStylePropAtom MetaCharAtom TextBookAtom TextSpecialInfoAtom
InteractiveAtom TextOrMaster)
OutlineAtom = RT_OutlineTextRefAtom [RT_TextRulerAtom]
TextStylePropAtom = *1(RT_StyleTextPropAtom / RT_MasterTextPropAtom)
MetaCharAtom = *(RT_SlideNumberMetaCharAtom / RT_DateTimeMetaCharAtom /
RT_GenericDateMetaCharAtom / RT_HeaderMetaCharAtom / RT_FooterMetaCharAtom /
RT_RtfDateTimeMetaCharAtom)
TextBookAtom = *RT_TextBookmarkAtom
TextSpecialInfoAtom = *1RT_TextSpecialInfoAtom
InteractiveAtom = *(RT_InteractiveInfo RT_TextInteractiveInfoAtom)
TextOrMaster = *(RT_TextRulerAtom / RT_MasterTextPropAtom)
```

2.9.77 TextClientDataSubContainerOrAtom

Referenced by: OfficeArtClientTextbox

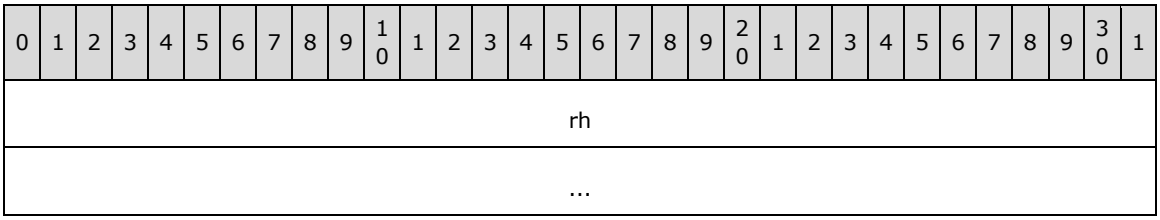
A variable type record whose type and meaning are dictated by the value of **rh.recType**, as specified in the following table.

Value	Meaning
RT_OutlineTextRefAtom	An OutlineTextRefAtom record that specifies a reference to a TextHeaderAtom record contained in the **SlideListWithTextContainer** record (section 2.4.14.3). The TextHeaderAtom record specifies the text for the shape that contains this record.
RT_TextHeaderAtom	A TextHeaderAtom record that specifies the type of a body of text.
RT_TextCharsAtom	A TextCharsAtom record that specifies text characters.
RT_TextBytesAtom	A TextBytesAtom record that specifies text characters.
RT_StyleTextPropAtom	A StyleTextPropAtom record that specifies text character and paragraph properties.
RT_SlideNumberMetaCharAtom	A SlideNumberMCAtom record that specifies a slide number metacharacter.
RT_DateTimeMetaCharAtom	A DateTimeMCAtom record that specifies a datetime metacharacter.
RT_GenericDateMetaCharAtom	A GenericDateMCAtom record that specifies a generic date metacharacter.
RT_HeaderMetaCharAtom	A HeaderMCAtom record that specifies a header metacharacter.
RT_FooterMetaCharAtom	A FooterMCAtom record that specifies a footer metacharacter.
RT_RtfDateTimeMetaCharAtom	A RTFDateTimeMCAtom record that specifies an RTF datetime metacharacter.
RT_TextBookmarkAtom	A TextBookmarkAtom record that specifies a text bookmark (1).
RT_TextSpecialInfoAtom	A TextSpecialInfoAtom record that specifies additional text properties.
RT_InteractiveInfo	An InteractiveInfoInstance record that specifies text interactive information.
RT_TextInteractiveInfoAtom	A TextInteractiveInfoInstance record that specifies the anchor for text interactive information.
RT_TextRulerAtom	A TextRulerAtom record that specifies a text ruler.
RT_MasterTextPropAtom	A MasterTextPropAtom record that specifies style properties for text on a master slide.

2.9.78 OutlineTextRefAtom

Referenced by: TextClientDataSubContainerOrAtom

An atom record that specifies a reference to text contained in the **SlideListWithTextContainer** record (section 2.4.14.3). Let the *corresponding slide persist* be specified by the **SlidePersistAtom** record (section 2.4.14.5) contained in the **SlideListWithTextContainer** record whose **persistIdRef** field refers to the **SlideContainer** record (section 2.5.1) that contains this **OutlineTextRefAtom** record.



index

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_OutlineTextRefAtom.
rh.recLen	MUST be 0x00000004.

index (4 bytes): A **signed integer** that specifies a zero-based index into the sequence of TextHeaderAtom records that follows the *corresponding slide persist*. It MUST be greater than or equal to 0x00000000 and less than the number of TextHeaderAtom records that follow the *corresponding slide persist*.

2.9.79 MasterTextPropAtom

Referenced by: TextClientDataSubContainerOrAtom

An atom record that specifies the indent levels for the text.

Let the *corresponding text* be specified by the TextHeaderAtom record that most closely precedes this **MasterTextPropAtom** record.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
rgMasterTextPropRun (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_MasterTextPropAtom.

rgMasterTextPropRun (variable): An array of MasterTextPropRun structures that specifies indent levels. The **count** field of each MasterTextPropRun specifies the number of characters to which the indent level applies, starting with the character at the zero-based index equal to the sum of the **count** fields of all previous MasterTextPropRun items in the array.

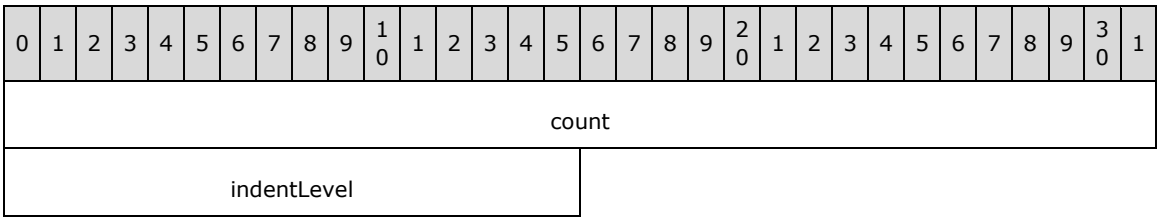
The sum of the **count** fields of the array items MUST be equal to the number of characters in the *corresponding text*. The length, in bytes, of the array is specified by **rh.recLen**.

2.9.80 MasterTextPropRun

Referenced by: MasterTextPropAtom

A structure that specifies the indent level for a text run.

Let the *corresponding text* be as specified in the MasterTextPropAtom record that contains this **MasterTextPropRun** structure.



count (4 bytes): An unsigned integer that specifies the number of characters of the corresponding text to which **indentLevel** applies.

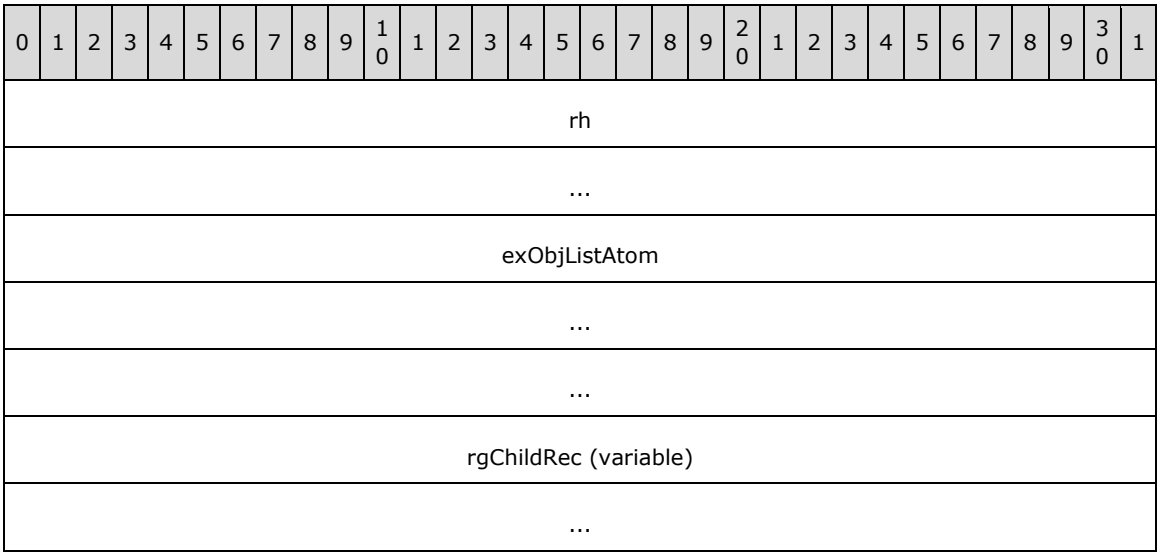
indentLevel (2 bytes): An IndentLevel that specifies the indent level of the characters.

2.10 External Object Types

2.10.1 ExObjListContainer

Referenced by: DocumentContainer

A container record that specifies a list of external objects in the document.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.

Field	Meaning
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalObjectList.

exObjListAtom (12 bytes): An ExObjListAtom record that specifies list-specific properties.

rgChildRec (variable): An array of ExObjListSubContainer records that specifies the external objects. The length, in bytes, of the array is specified by the following formula:

```
rh.recLen - 12.
```

2.10.2 ExObjListSubContainer

Referenced by: ExObjListContainer

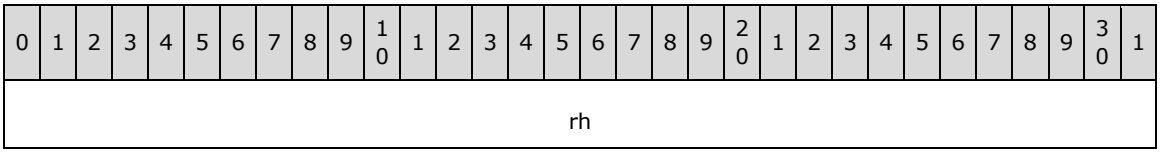
A variable type record whose type and meaning are dictated by the value of **rh.recType**, as specified in the following table.

Value	Meaning
RT_ExternalAviMovie	An ExAviMovieContainer record that specifies an **Audio Video Interleaved (AVI)** movie.
RT_ExternalCdAudio	An ExCDAudioContainer record that specifies information about CD audio.
RT_ExternalOleControl	An **ExControlContainer** record (section 2.10.10) that specifies an ActiveX control.
RT_ExternalHyperlink	An ExHyperlinkContainer record that specifies information about a hyperlink.
RT_ExternalMciMovie	An ExMCIMovieContainer record that specifies an **MCI** movie.
RT_ExternalMidiAudio	An ExMIDIAudioContainer record that specifies information about MIDI audio.
RT_ExternalOleEmbed	An **ExOleEmbedContainer** record (section 2.10.27) that specifies an embedded OLE object.
RT_ExternalOleLink	An **ExOleLinkContainer** record (section 2.10.29) that specifies a linked OLE object.
RT_ExternalWavAudioEmbedded	An ExWAVAudioEmbeddedContainer record that specifies an embedded WAV sound.
RT_ExternalWavAudioLink	An ExWAVAudioLinkContainer record that specifies a linked WAV sound.

2.10.3 ExObjListAtom

Referenced by: ExObjListContainer

An atom record that specifies properties for the containing **ExObjListContainer** record (section 2.10.1).



...
exObjIdSeed

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalObjectListAtom.
rh.recLen	MUST be 0x00000004.

exObjIdSeed (4 bytes): A signed integer that specifies a seed for creating a new **ExObjId** (section 2.2.7) or **ExHyperlinkId** (section 2.2.5) value. It MUST be greater than or equal to the largest **ExObjId** value in the file as specified by the **ExMediaAtom** record (section 2.10.6) or **ExOleObjAtom** record (section 2.10.12) and MUST be greater than or equal to the largest **ExHyperlinkId** value in the file as specified by the **ExHyperlinkAtom** record (section 2.10.17). It MUST be greater than or equal to 0x00000001.

2.10.4 ExAviMovieContainer

Referenced by: ExObjListSubContainer

A container record that specifies information about an AVI movie.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
exVideo (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

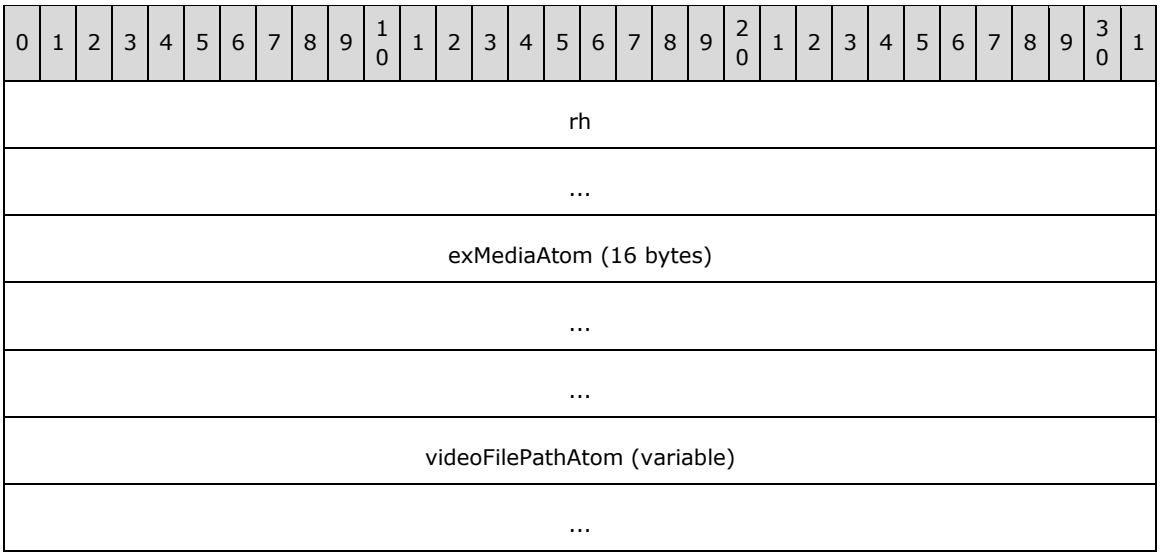
Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalAviMovie.

exVideo (variable): An ExVideoContainer record that specifies information about the AVI movie.

2.10.5 ExVideoContainer

Referenced by: ExAviMovieContainer, ExMCIMovieContainer

A container record that specifies information about external video data.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalVideo.

exMediaAtom (16 bytes): An **ExMediaAtom** record (section 2.10.6) that specifies information about the external video.

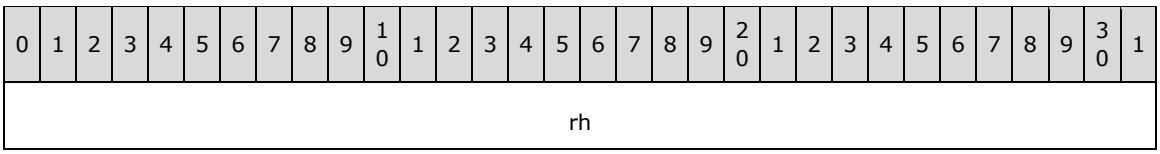
videoFilePathAtom (variable): An optional UncOrLocalPathAtom record that specifies the UNC or local path to a video file. The length, in bytes, of the field is specified by the following formula:

```
rh.recLen - 16.
```

2.10.6 ExMediaAtom

Referenced by: ExCDAudioContainer, ExMIDIAudioContainer, ExVideoContainer, ExWAVAudioEmbeddedContainer, ExWAVAudioLinkContainer

An atom record that specifies information about external audio or video data.



...				
exObjId				
A	B	C	reserved	unused

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalMediaAtom.
rh.recLen	MUST be 0x00000008.

exObjId (4 bytes): An **ExObjId** (section 2.2.7) that specifies the identifier for the audio or video data.

A - fLoop (1 bit): A bit that specifies whether the audio or video data is repeated continuously during playback.

B - fRewind (1 bit): A bit that specifies whether the audio or video data is rewound after playing.

C - fNarration (1 bit): A bit that specifies whether the audio data is recorded narration for the slide show. It MUST be **FALSE** if this **ExMediaAtom** record is contained by an ExVideoContainer record.

reserved (13 bits): MUST be zero and MUST be ignored.

unused (2 bytes): Undefined and MUST be ignored.

2.10.7 UncOrLocalPathAtom

Referenced by: ExMIDIAudioContainer, ExVideoContainer, ExWAVAudioLinkContainer

An atom record that specifies a UNC or local path to a file.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
path (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

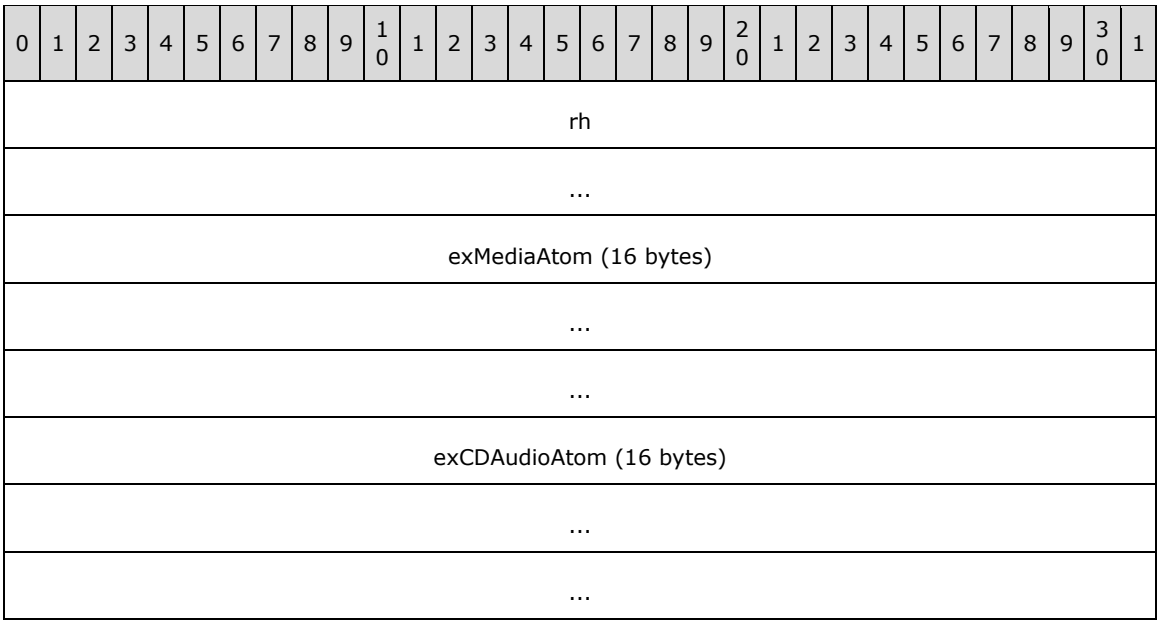
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number.

path (variable): A UncOrLocalPath that specifies the UNC or local path to a file. The length, in bytes, of the field is specified by **rh.recLen**.

2.10.8 ExCDAudioContainer

Referenced by: ExObjListSubContainer

A container record that specifies information about compact disc (CD) audio.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalCdAudio.

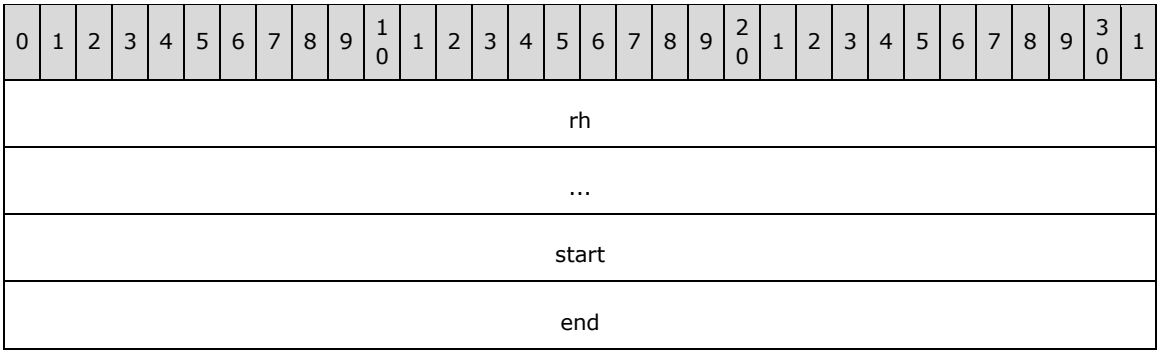
exMediaAtom (16 bytes): An **ExMediaAtom** record (section 2.10.6) that specifies information about the CD audio.

exCDAudioAtom (16 bytes): An ExCDAudioAtom record that specifies start and end information for the CD audio.

2.10.9 ExCDAudioAtom

Referenced by: ExCDAudioContainer

An atom record that specifies start and end information for a CD audio clip.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalCdAudioAtom.
rh.recLen	MUST be 0x00000008.

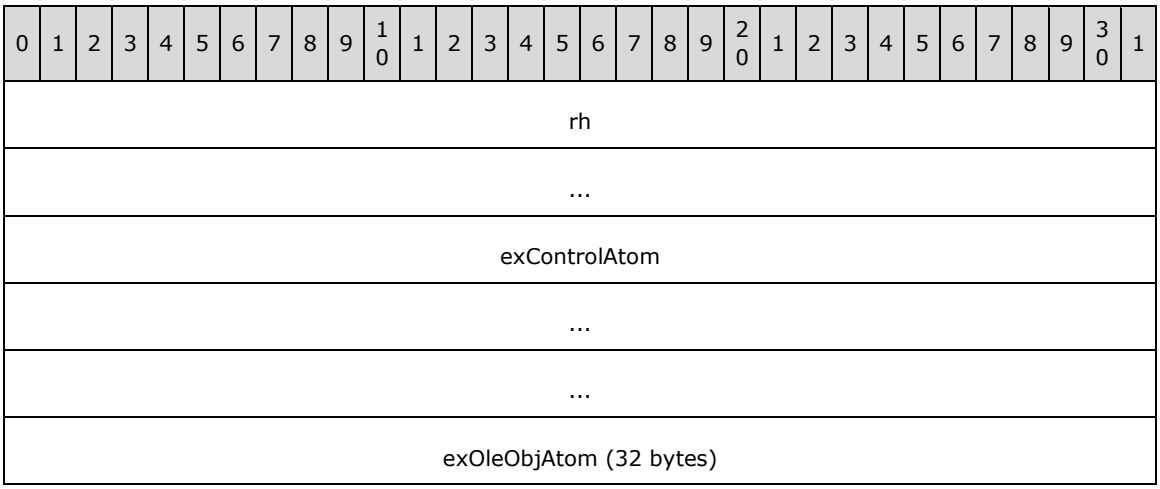
start (4 bytes): A TmsfTimeStruct structure that specifies the beginning of the CD audio clip. It MUST be less than or equal to the value specified by **end**.

end (4 bytes): A TmsfTimeStruct structure that specifies the end of the CD audio clip. It MUST be greater than or equal to the value specified by **start**.

2.10.10 ExControlContainer

Referenced by: ExObjListSubContainer

A container record that specifies information about an ActiveX control.



...
...
menuNameAtom (variable)
...
progIdAtom (variable)
...
clipboardNameAtom (variable)
...
metafile (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalOleControl.

exControlAtom (12 bytes): An ExControlAtom record that specifies the identifying information for this ActiveX control.

exOleObjAtom (32 bytes): An **ExOleObjAtom** record (section 2.10.12) that specifies information about this ActiveX control as an OLE object.

menuNameAtom (variable): An optional MenuNameAtom record that specifies the name to use in the user interface.

progIdAtom (variable): An optional ProgIDAtom record that specifies the **ProgID** (described in [MSDN-COM]) of the ActiveX control.

clipboardNameAtom (variable): An optional ClipboardNameAtom record that specifies the name used by the user interface during copy and paste operations.

metafile (variable): An optional MetafileBlob record that specifies the icon for the ActiveX control.

2.10.11 ExControlAtom

Referenced by: ExControlContainer

An atom record that specifies an ActiveX control.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
slideIdRef																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalOleControlAtom.
rh.recLen	MUST be 0x00000004.

slideIdRef (4 bytes): A **SlideIdRef** (section 2.2.25) that specifies which presentation slide is associated with the ActiveX control.

2.10.12 ExOleObjAtom

Referenced by: ExControlContainer, ExOleEmbedContainer, ExOleLinkContainer

An atom record that specifies information about OLE objects. Each **ExOleObjAtom** MUST be referred to by exactly one ExObjRefAtom.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
drawAspect																															
type																															
exObjId																															
subType																															
persistIdRef																															
unused																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x1.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalOleObjectAtom.
rh.recLen	MUST be 0x00000018.

drawAspect (4 bytes): A **DataViewAspectEnum** ([MS-OSHARED] section 2.2.1.2) that specifies the view aspect used to display the OLE object.

type (4 bytes): An **ExOleObjTypeEnum** enumeration that specifies the type of OLE object.

exObjId (4 bytes): An **ExObjId** (section 2.2.7) that specifies a unique identifier for the OLE object.

subType (4 bytes): An **ExOleObjSubTypeEnum** enumeration that specifies the sub-type of the OLE object.

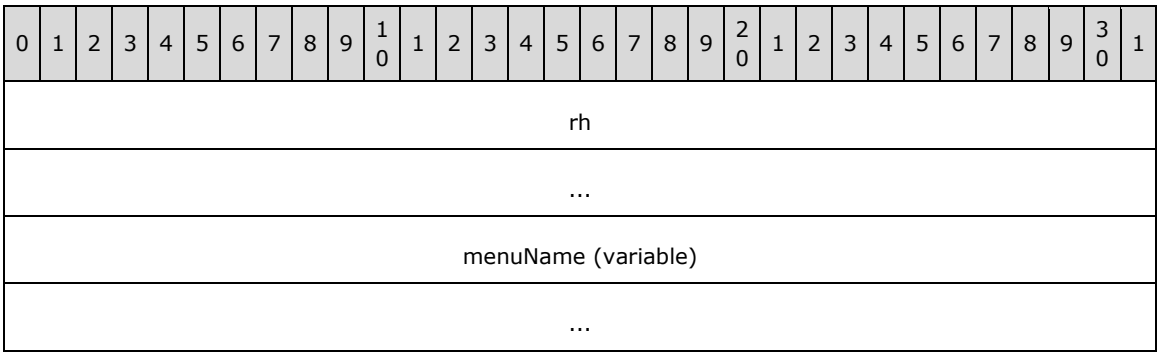
persistIdRef (4 bytes): A **PersistIdRef** (section 2.2.21) that specifies the value to look up in the persist object directory to find the offset of an **ExOleObjStg** (section 2.10.34) or an **ExControlStg** (section 2.10.37).

unused (4 bytes): Undefined and MUST be ignored.

2.10.13 MenuNameAtom

Referenced by: ExControlContainer, ExOleEmbedContainer, ExOleLinkContainer

An atom record that specifies the short name of an OLE object or an ActiveX control.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

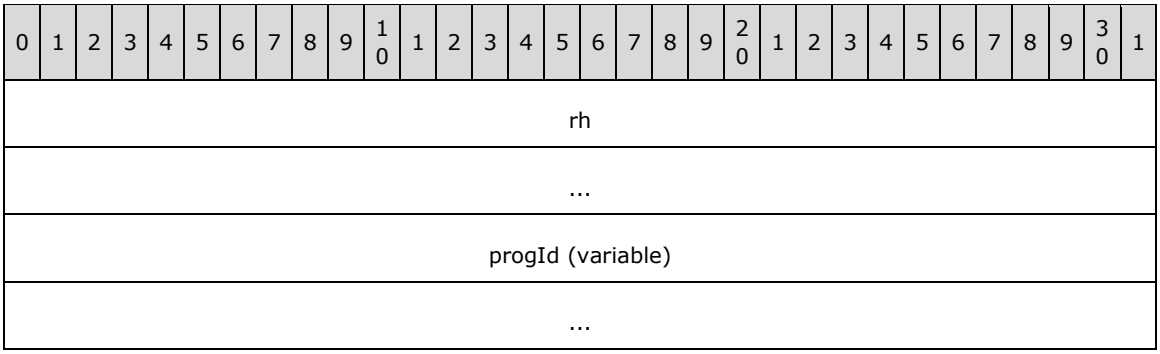
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be an **RT_CString** (section 2.13.24).
rh.recLen	MUST be even.

menuName (variable): A UnicodeString that specifies the name. The length, in bytes, of the field is specified by **rh.recLen**.

2.10.14 ProgIDAtom

Referenced by: ExControlContainer, ExOleEmbedContainer, ExOleLinkContainer

An atom record that specifies the programmatic identifier of an OLE object or an ActiveX control. A **ProgID** (described in [MSDN-COM]), a programmatic identifier, is a string that uniquely identifies a class.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

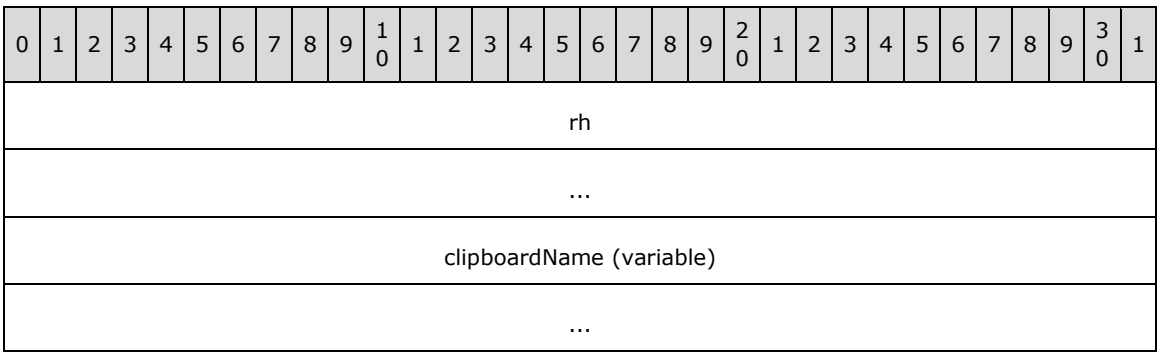
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x002.
rh.recType	MUST be an **RT_CString** (section 2.13.24).
rh.recLen	MUST be even.

progId (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the **ProgID**. The length, in bytes, of the field is specified by **rh.recLen**.

2.10.15 ClipboardNameAtom

Referenced by: ExControlContainer, ExOleEmbedContainer, ExOleLinkContainer

An atom record that specifies the full descriptive class name of an OLE object or an ActiveX control.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

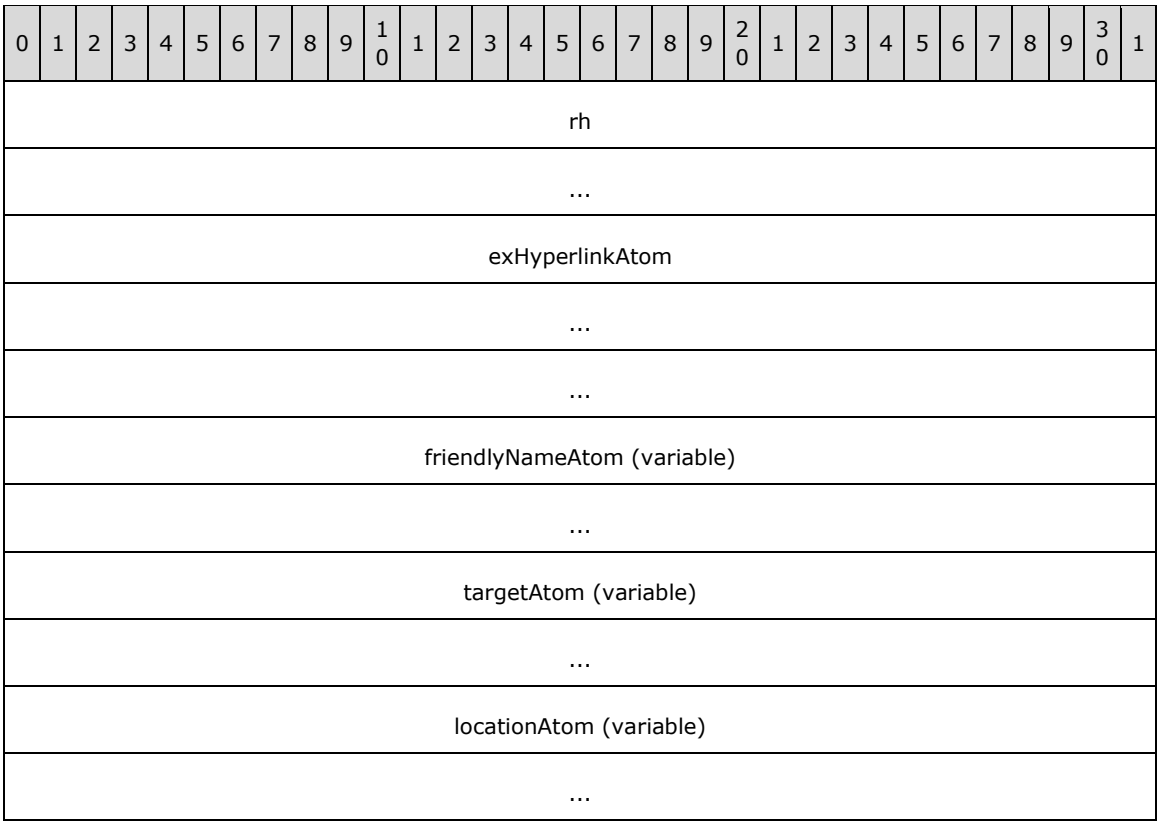
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x003.
rh.recType	MUST be an **RT_CString** (section 2.13.24).
rh.recLen	MUST be even.

clipboardName (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the full descriptive class name. The length, in bytes, of the field is specified by **rh.recLen**.

2.10.16 ExHyperlinkContainer

Referenced by: ExObjListSubContainer

A container record that specifies information about a hyperlink.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalHyperlink.

exHyperlinkAtom (12 bytes): An **ExHyperlinkAtom** record (section 2.10.17) that specifies information needed to identify this hyperlink.

friendlyNameAtom (variable): An optional FriendlyNameAtom record that specifies the user-readable name of this hyperlink.

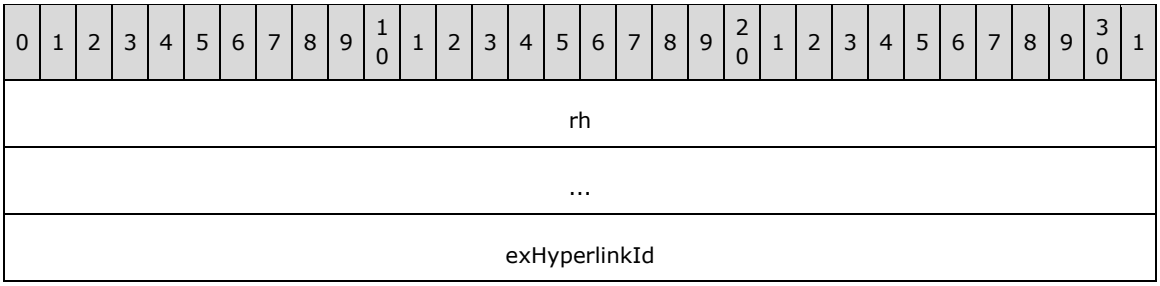
targetAtom (variable): An optional TargetAtom record that specifies the full path of the destination file of this hyperlink.

locationAtom (variable): An optional LocationAtom record that specifies the location within the destination file of the hyperlink.

2.10.17 ExHyperlinkAtom

Referenced by: ExHyperlinkContainer

An atom record that specifies the value needed to look up a hyperlink within the collection of external objects as specified by the **ExObjListContainer** record (section 2.10.1).



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

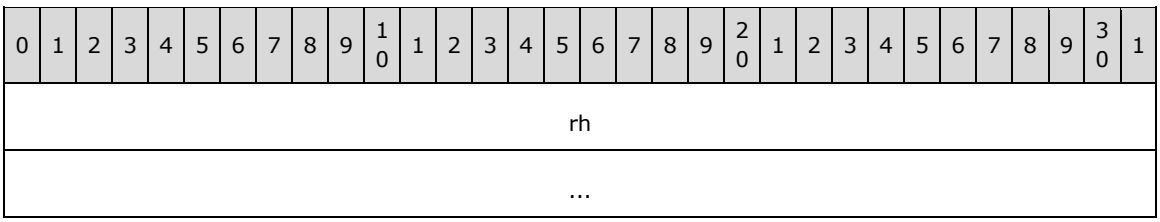
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalHyperlinkAtom.
rh.recLen	MUST be 0x00000004.

exHyperlinkId (4 bytes): An **ExHyperlinkId** (section 2.2.5) that specifies the identifier of this hyperlink.

2.10.18 FriendlyNameAtom

Referenced by: ExHyperlinkContainer

An atom record that specifies the user-readable name of a hyperlink.



friendlyName (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an **RT_CString** (section 2.13.24).
rh.recLen	MUST be even.

friendlyName (variable): A UnicodeString that specifies the user-readable name. The length, in bytes, of the field is specified by **rh.recLen**.

2.10.19 TargetAtom

Referenced by: ExHyperlinkContainer

An atom record that specifies the full path of the hyperlink destination file.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
target (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

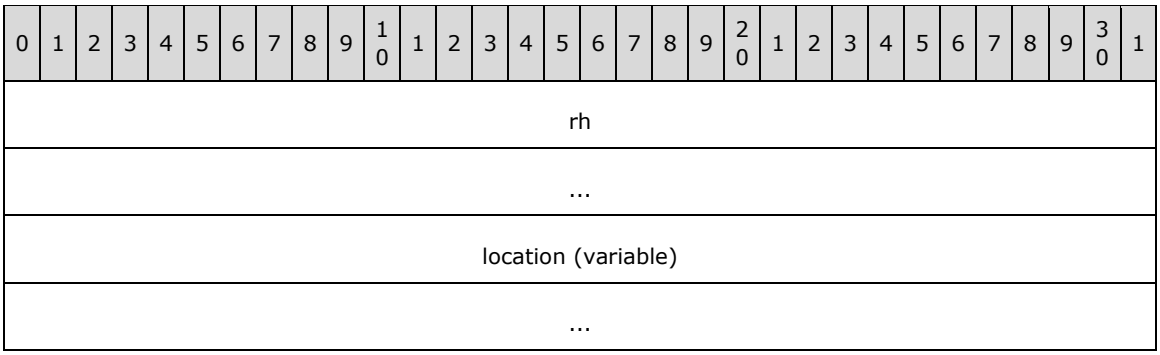
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be an **RT_CString** (section 2.13.24).
rh.recLen	MUST be even.

target (variable): A UnicodeString that specifies the full path of the destination file. The length, in bytes, of the field is specified by **rh.recLen**.

2.10.20 LocationAtom

Referenced by: ExHyperlinkContainer

An atom record that specifies the location of the hyperlink within the destination file.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

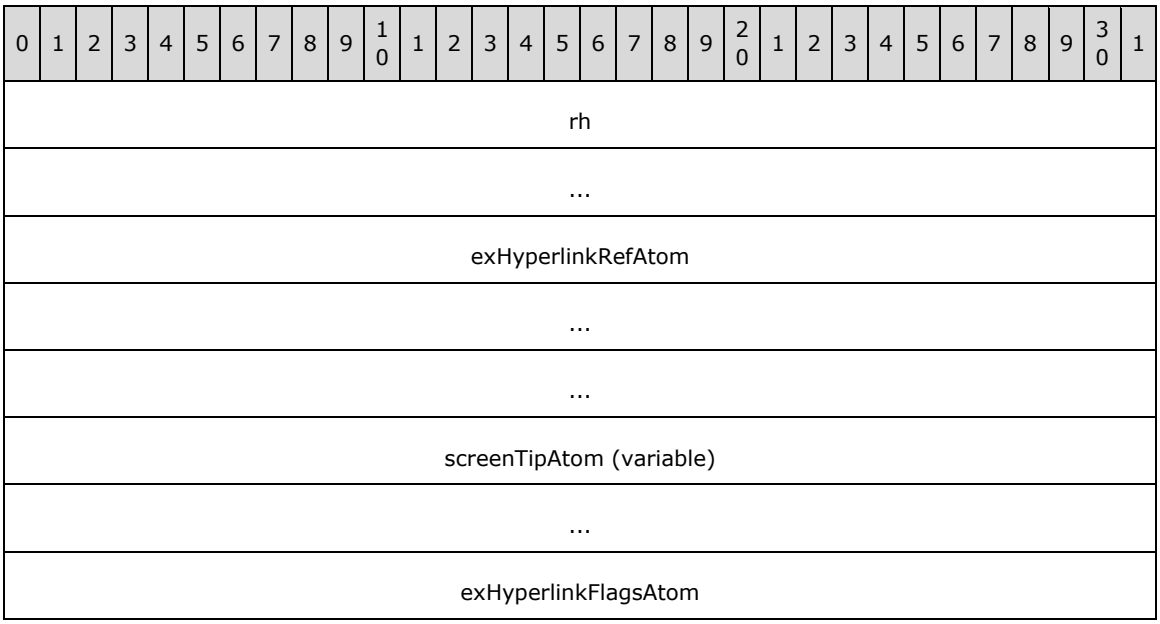
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x003.
rh.recType	MUST be an **RT_CString** (section 2.13.24).
rh.recLen	MUST be even.

location (variable): A UnicodeString that specifies the location within the destination file. The length, in bytes, of the field is specified by **rh.recLen**.

2.10.21 ExHyperlink9Container

Referenced by: PP9DocBinaryTagExtension

A container record that specifies additional information about a hyperlink.



...
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalHyperlink9.

exHyperlinkRefAtom (12 bytes): An ExHyperlinkRefAtom record that specifies a reference to the corresponding hyperlink.

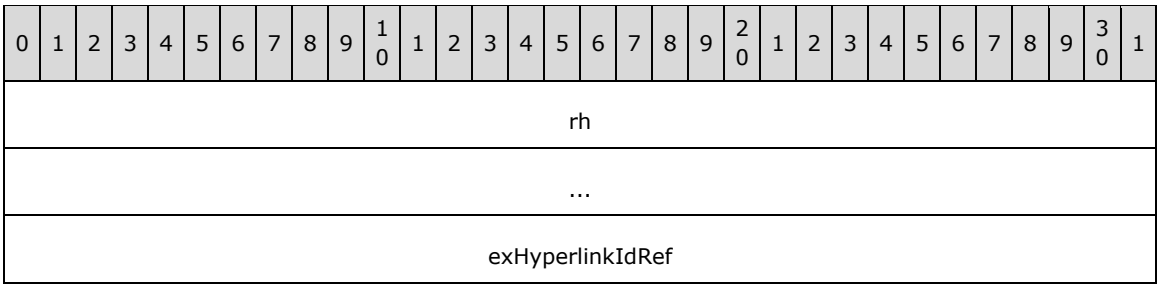
screenTipAtom (variable): An optional ScreenTipAtom record that specifies the screen tip of the corresponding hyperlink.

exHyperlinkFlagsAtom (12 bytes): An ExHyperlinkFlagsAtom record that specifies additional information about the corresponding hyperlink.

2.10.22 ExHyperlinkRefAtom

Referenced by: ExHyperlink9Container

An atom record that specifies a reference to a hyperlink.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

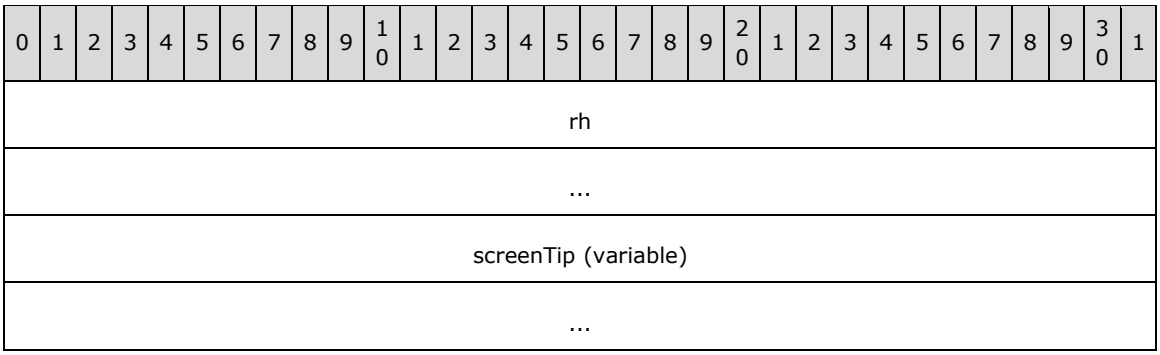
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalHyperlinkAtom.
rh.recLen	MUST be 0x00000004.

exHyperlinkIdRef (4 bytes): An ExHyperlinkIdRef that specifies an identifier that references the corresponding hyperlink.

2.10.23 ScreenTipAtom

Referenced by: ExHyperlink9Container

An atom record that specifies the screen tip of a hyperlink.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

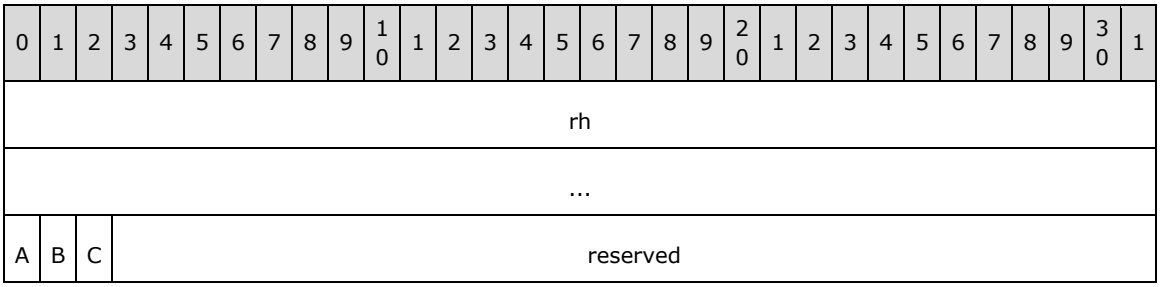
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an **RT_CString** (section 2.13.24).
rh.recLen	MUST be even.

screenTip (variable): A UnicodeString that specifies the screen tip. The length, in bytes, of the field is specified by **rh.recLen**.

2.10.24 ExHyperlinkFlagsAtom

Referenced by: ExHyperlink9Container

An atom record that specifies additional information about a hyperlink.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalHyperlinkFlagsAtom.
rh.recLen	MUST be 0x00000004.

A - fInsertHyperlinkDialog (1 bit): A bit that specifies whether this hyperlink was created in the **Insert Hyperlink** dialog box.

B - fLocationIsNamedShow (1 bit): A bit that specifies whether the location of this hyperlink is a named show.

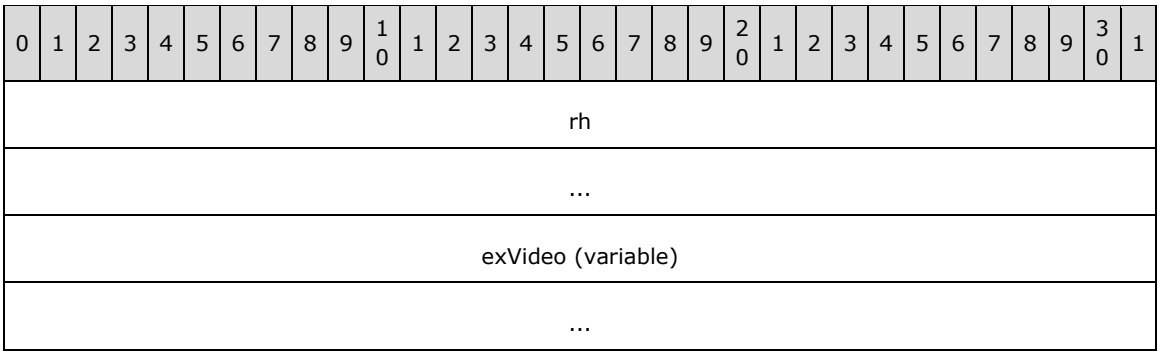
C - fNamedShowReturnToSlide (1 bit): A bit that specifies whether this hyperlink to a named show was set to return to the slide.

reserved (29 bits): MUST be zero and MUST be ignored.

2.10.25 ExMCIMovieContainer

Referenced by: ExObjListSubContainer

A container record that specifies information about a movie stored externally as a Media Control Interface (MCI) file.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

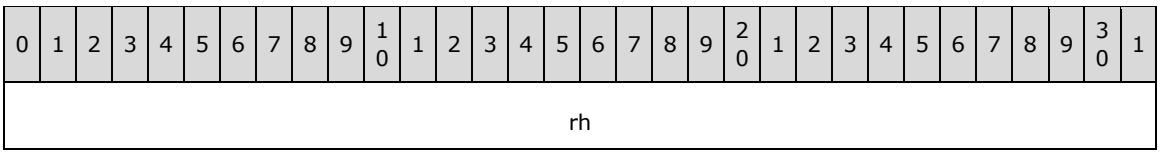
Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalMciMovie.

exVideo (variable): An ExVideoContainer record that specifies information about the MCI movie.

2.10.26 ExMIDIAudioContainer

Referenced by: ExObjListSubContainer

A container record that specifies information about audio stored externally as a **Musical Instrument Digital Interface (MIDI)** file.



...
exMediaAtom (16 bytes)
...
...
audioFilePathAtom (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalMidiAudio.

exMediaAtom (16 bytes): An **ExMediaAtom** record (section 2.10.6) that specifies information about the external MIDI audio.

audioFilePathAtom (variable): An optional UncOrLocalPathAtom record that specifies the UNC or local path to a MIDI audio file. The length, in bytes, of the field is specified by the following formula:

```
rh.recLen - 16.
```

2.10.27 ExOleEmbedContainer

Referenced by: ExObjListSubContainer

A container record that specifies information about an embedded OLE object.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
exOleEmbedAtom (16 bytes)																															
...																															
...																															
exOleObjAtom (32 bytes)																															

...
...
menuNameAtom (variable)
...
progIdAtom (variable)
...
clipboardNameAtom (variable)
...
metafile (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalOleEmbed.

exOleEmbedAtom (16 bytes): An ExOleEmbedAtom record that specifies information about embedding an OLE object.

exOleObjAtom (32 bytes): An **ExOleObjAtom** record (section 2.10.12) that specifies information about the OLE object.

menuNameAtom (variable): An optional MenuNameAtom record that specifies the name to use in the user interface.

progIdAtom (variable): An optional ProgIDAtom record that specifies the **ProgID** (described in [MSDN-COM]) of the OLE object.

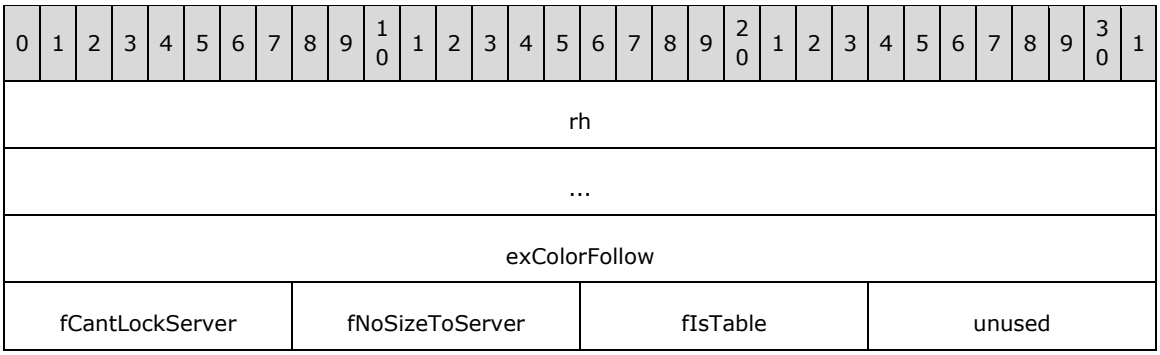
clipboardNameAtom (variable): An optional ClipboardNameAtom record that specifies the name used by the user interface during copy and paste operations.

metafile (variable): An optional MetafileBlob record that specifies the icon for the OLE object.

2.10.28 ExOleEmbedAtom

Referenced by: ExOleEmbedContainer

An atom record that specifies preferences for embedding an OLE object.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalOleEmbedAtom.
rh.recLen	MUST be 0x00000008.

exColorFollow (4 bytes): An **ExColorFollowEnum** enumeration that specifies how the OLE object follows the color scheme of the containing document.

fCantLockServer (1 byte): A **bool1** (section 2.2.2) that specifies whether the OLE server for the embedded OLE object cannot be locked.

fNoSizeToServer (1 byte): A byte that specifies whether sending the OLE object dimensions to the OLE server can be omitted. It SHOULD<108> be a value from the following table.

Value	Meaning
0x00	Do send the dimensions to the OLE server.
0x01	Do not send the dimension to the OLE server.

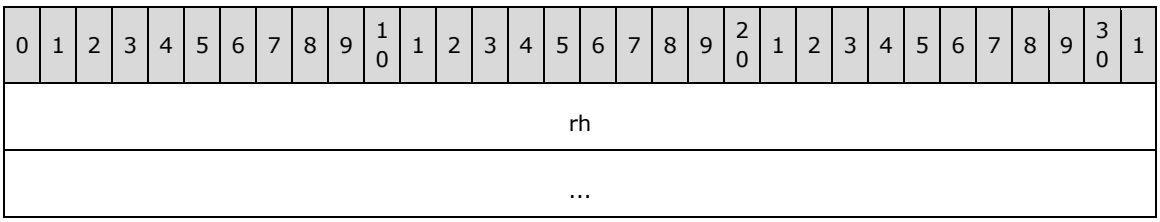
fIsTable (1 byte): A **bool1** that specifies whether the OLE object represents a table created by **ProgID** (described in [MSDN-COM]) **Word.Document**. It SHOULD<109> be ignored.

unused (1 byte): Undefined and MUST be ignored.

2.10.29 ExOleLinkContainer

Referenced by: ExObjListSubContainer

A container record that specifies information about a linked OLE object.



exOleLinkAtom (20 bytes)
...
...
exOleObjAtom (32 bytes)
...
...
menuNameAtom (variable)
...
progIdAtom (variable)
...
clipboardNameAtom (variable)
...
metafile (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalOleLink.

exOleLinkAtom (20 bytes): An ExOleLinkAtom record that specifies OLE object linking information.

exOleObjAtom (32 bytes): An **ExOleObjAtom** record (section 2.10.12) that specifies information about the linked OLE object.

menuNameAtom (variable): An optional MenuNameAtom record that specifies the name to use in the user interface.

progIdAtom (variable): An optional ProgIDAtom record that specifies the **ProgID** (described in [MSDN-COM]) of the OLE object.

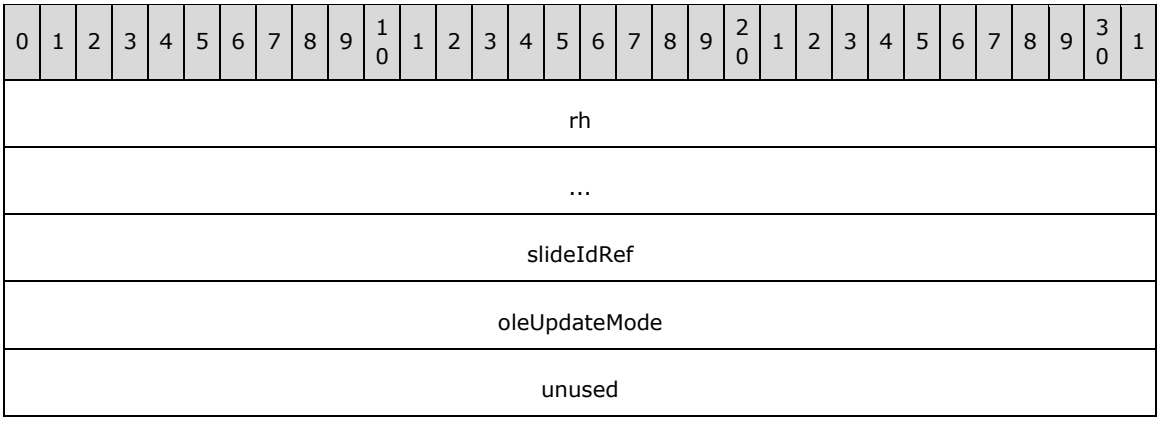
clipboardNameAtom (variable): An optional ClipboardNameAtom record that specifies the name used by the user interface during copy and paste operations.

metafile (variable): An optional MetafileBlob record that specifies the icon for the OLE object.

2.10.30 ExOleLinkAtom

Referenced by: ExOleLinkContainer

An atom record that specifies information about a linked OLE object.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalOleLinkAtom.
rh.recLen	MUST be 0x0000000C.

slideIdRef (4 bytes): A **SlideIdRef** (section 2.2.25) that specifies the presentation slide associated with this OLE object.

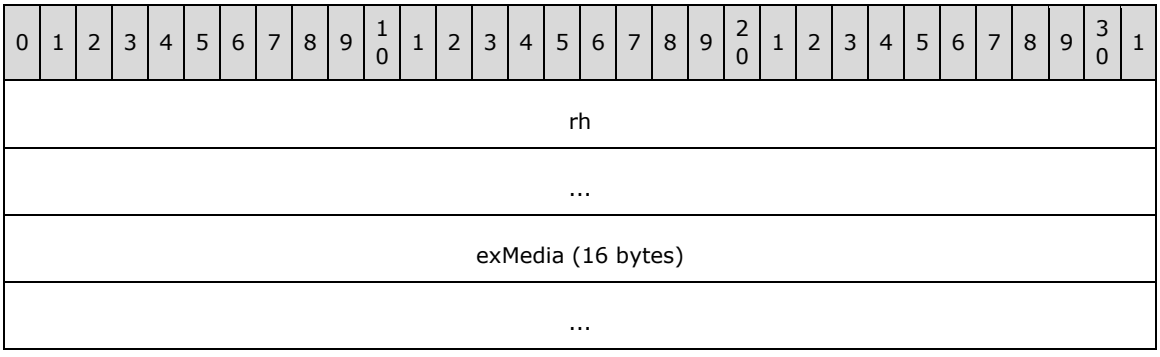
oleUpdateMode (4 bytes): An **ObjectUpdateEnum** ([MS-OSHARED] section 2.2.1.1) that specifies how the OLE object is updated.

unused (4 bytes): Undefined and MUST be ignored.

2.10.31 ExWAVAudioEmbeddedContainer

Referenced by: ExObjListSubContainer

A container record that specifies information about embedded WAV audio.



...
exWavAudioEmbedded (16 bytes)
...
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalWavAudioEmbedded.
rh.recLen	MUST be 0x20.

exMedia (16 bytes): An **ExMediaAtom** record (section 2.10.6) that specifies information about the WAV audio.

exWavAudioEmbedded (16 bytes): An ExWAVAudioEmbeddedAtom record that specifies information about an embedded WAV audio in the **SoundCollectionContainer** record (section 2.4.16.1).

2.10.32 ExWAVAudioEmbeddedAtom

Referenced by: ExWAVAudioEmbeddedContainer

An atom record that specifies information about an embedded WAV audio.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
soundIdRef																															
soundLength																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x1.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be an RT_ExternalWavAudioEmbeddedAtom.
rh.recLen	MUST be 0x00000008.

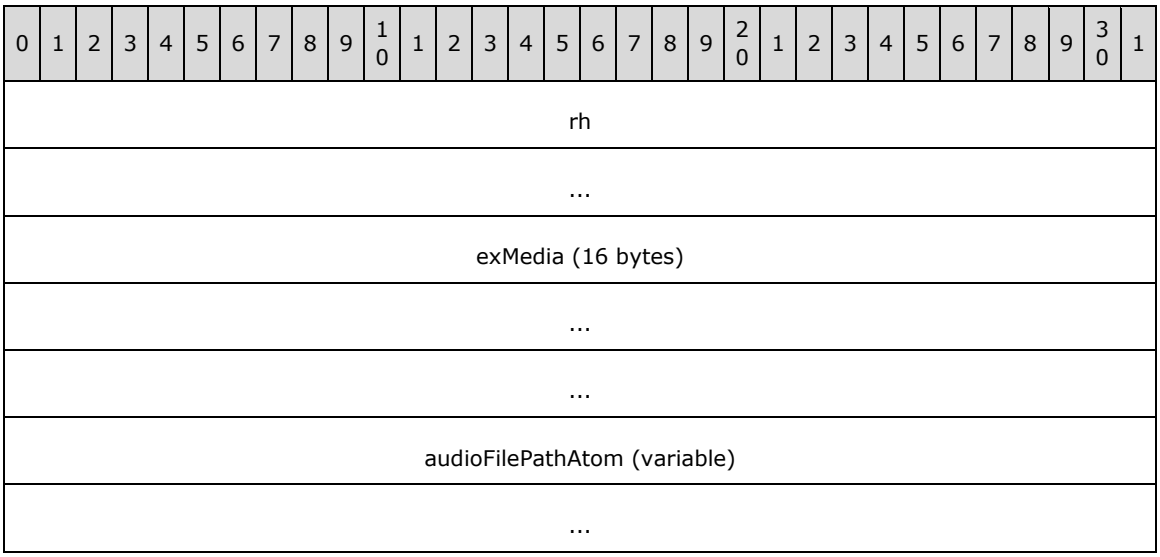
soundIdRef (4 bytes): A SoundIdRef that specifies the value to look up in the **SoundCollectionContainer** record (section 2.4.16.1) to find the embedded audio.

soundLength (4 bytes): A signed integer that specifies the duration, in milliseconds, for which to play the audio. It MUST be greater than or equal to 0x00000000.

2.10.33 ExWAVAudioLinkContainer

Referenced by: ExObjListSubContainer

A container record that specifies information about external WAV audio.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalWavAudioLink.

exMedia (16 bytes): An **ExMediaAtom** record (section 2.10.6) that specifies information about the external WAV audio.

audioFilePathAtom (variable): An optional UncOrLocalPathAtom record that specifies the UNC or local path to a WAV audio file. The length, in bytes, of the field is specified by the following formula:

```
rh.recLen - 16.
```

2.10.34 ExOleObjStg

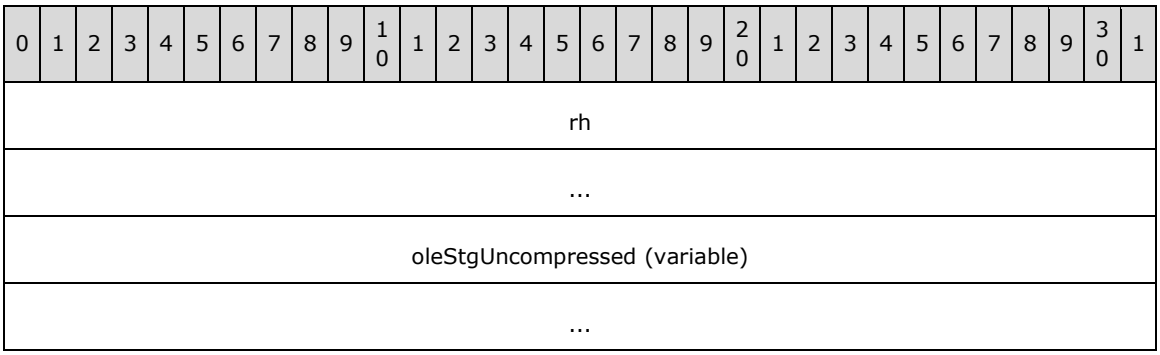
A variable type record whose type and meaning are specified by the value of the **rh.recInstance** field of the contained storage, as specified in the following table.

Value	Meaning
0x000	An ExOleObjStgUncompressedAtom record that specifies an uncompressed OLE object.
0x001	An ExOleObjStgCompressedAtom record that specifies a compressed OLE object.

2.10.35 ExOleObjStgUncompressedAtom

Referenced by: ExOleObjStg

An atom record that specifies an uncompressed storage of an OLE object.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

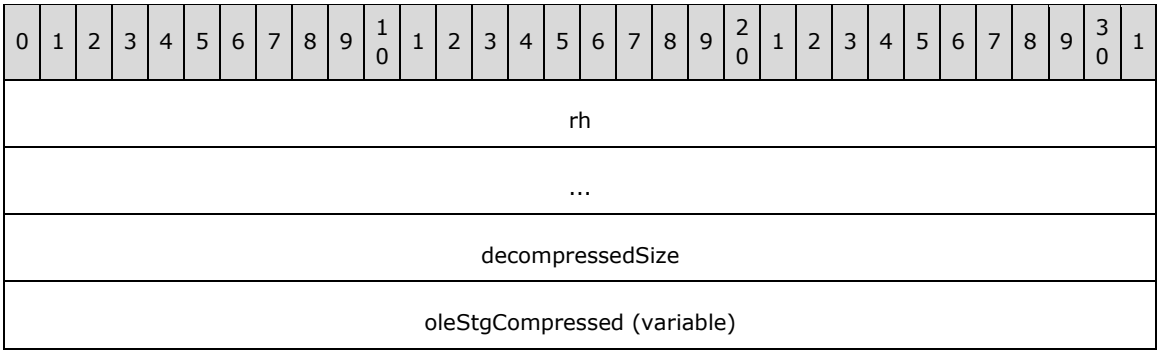
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalOleObjectStg.

oleStgUncompressed (variable): An array of bytes that specifies a structured storage (described in [MSDN-COM]) for the OLE object. The length, in bytes, of the field is specified by **rh.recLen.**

2.10.36 ExOleObjStgCompressedAtom

Referenced by: ExOleObjStg

An atom record that specifies a compressed storage for an OLE object.



...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be an RT_ExternalOleObjectStg.

decompressedSize (4 bytes): An **unsigned integer** that specifies the length, in bytes, of the storage after decompression.

oleStgCompressed (variable): An array of bytes that specifies a compressed structured storage (described in [MSDN-COM]) for the OLE object. The original bytes of the storage are compressed by the algorithm specified in [RFC1950] and are decompressed by the algorithm specified in [RFC1951]. The length, in bytes, of the field is specified by the following formula:

```
rh.recLen - 4
```

2.10.37 ExControlStg

A variable type record whose type and meaning are specified by the value of the record header member **rh.recInstance** of the contained storage, as specified in the following table.

Value	Meaning
0x000	An ExControlStgUncompressedAtom record that specifies an uncompressed storage for an ActiveX control.
0x001	An ExControlStgCompressedAtom record that specifies a compressed storage for an ActiveX control.

2.10.38 ExControlStgUncompressedAtom

Referenced by: ExControlStg

An atom record that specifies information about the uncompressed storage of an ActiveX control.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
pptControlStgUncompressed (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

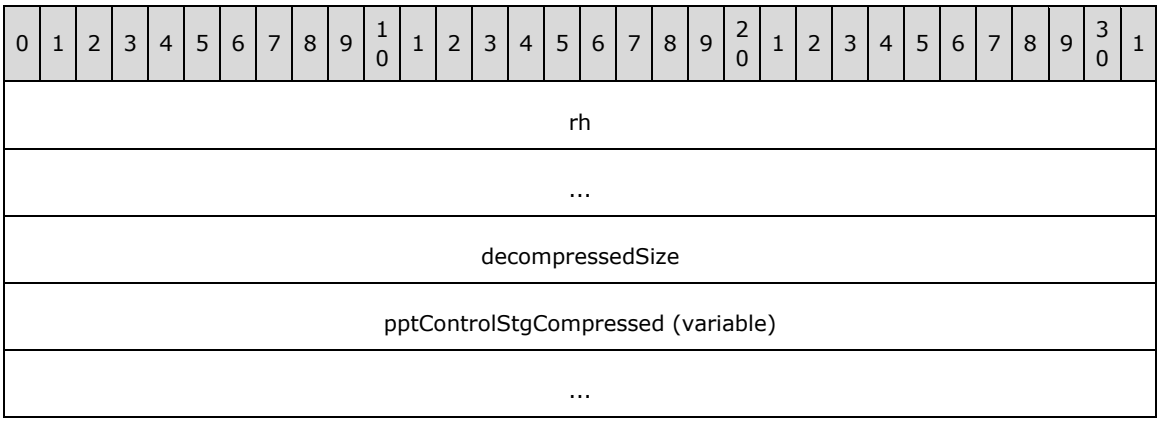
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalOleObjectStg.

pptControlStgUncompressed (variable): An array of bytes that specifies a structured storage (described in [MSDN-COM]) for the ActiveX control. Office Forms ActiveX controls are specified in [MS-OFORMS]. The length, in bytes, of the field is specified by **rh.recLen**.

2.10.39 ExControlStgCompressedAtom

Referenced by: ExControlStg

An atom record that specifies information about the compressed storage for an ActiveX control.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be an RT_ExternalOleObjectStg.

decompressedSize (4 bytes): An unsigned integer that specifies the length, in bytes, of the storage after decompression.

pptControlStgCompressed (variable): An array of bytes that specifies a compressed structured storage (described in [MSDN-COM]) for the ActiveX control. Office Forms ActiveX controls are specified in [MS-OFORMS]. The original bytes of the storage are compressed by the algorithm specified in [RFC1950] and are decompressed by the algorithm specified in [RFC1951]. The length, in bytes, of the field is specified by the following formula:

```
rh.recLen – 4.
```

2.10.40 VbaProjectStg

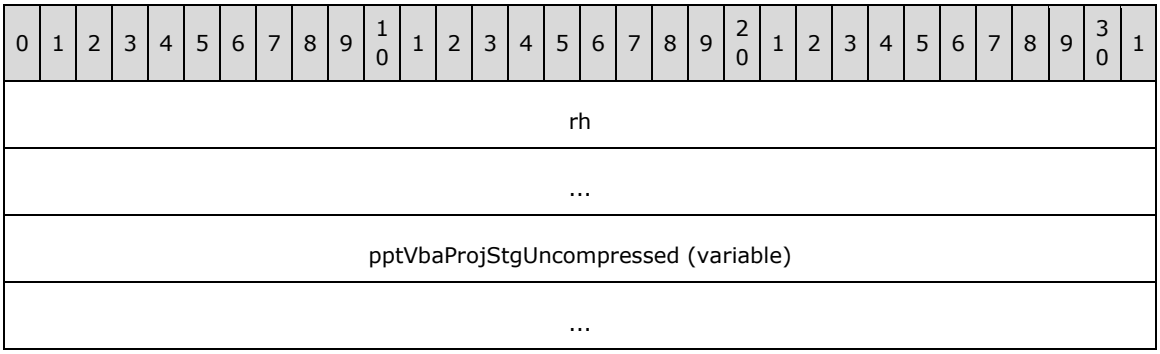
A variable type record whose type and meaning are dictated by the value of the record header member **rh.recInstance** of the contained storage, as specified in the following table.

Value	Meaning
0x000	A VbaProjectStgUncompressedAtom record that specifies uncompressed storage for a VBA project.
0x001	A VbaProjectStgCompressedAtom record that specifies compressed storage for a VBA project.

2.10.41 VbaProjectStgUncompressedAtom

Referenced by: VbaProjectStg

An atom record that specifies information about the uncompressed structured storage (described in [MSDN-COM]) for a VBA project.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

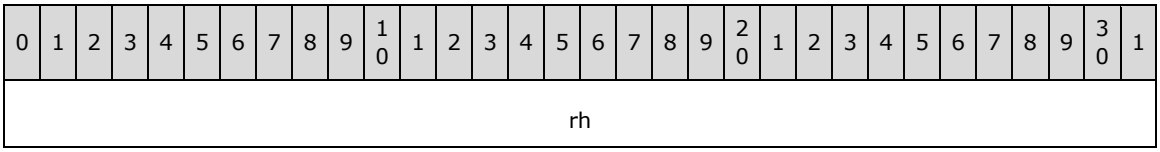
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ExternalOleObjectStg.

pptVbaProjStgUncompressed (variable): An array of bytes that specifies a structured storage (described in [MSDN-COM]) for the VBA project ([MS-OVBA] section 2.2.1). The length, in bytes, of the field is specified by **rh.recLen**.

2.10.42 VbaProjectStgCompressedAtom

Referenced by: VbaProjectStg

An atom record that specifies information about the compressed structured storage (described in [MSDN-COM]) for a VBA project.



...
decompressedSize
pptVbaProjStgCompressed (variable)
...

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be an RT_ExternalOleObjectStg.

decompressedSize (4 bytes): An unsigned integer that specifies the length of the storage after decompression.

pptVbaProjStgCompressed (variable): An array of bytes that specifies a compressed structured storage (described in [MSDN-COM]) for the VBA project as specified in [MS-OVBA] section 2.2.1. The original bytes of the storage are compressed by the algorithm specified in [RFC1950] and are decompressed by the algorithm specified in [RFC1951]. The length, in bytes, of the field is specified by the following formula:

```
rh.recLen - 4
```

2.11 Other Types

2.11.1 DocRoutingSlipAtom

Referenced by: DocumentContainer

An atom record that specifies information about a document **routing slip**.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
length																															
unused1																															
recipientCount																															
currentRecipient																															

A	B	C	D	E	F	reserved2
unused2						
originatorString (variable)						
...						
rgRecipientRoutingSlipStrings (variable)						
...						
subjectString (variable)						
...						
messageString (variable)						
...						
unused3 (variable)						
...						

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_DocRoutingSlipAtom.

length (4 bytes): An unsigned integer that specifies the length, in bytes, of the data contained in the document routing slip, including this field. It MUST be less than or equal to **rh.recLen**.

unused1 (4 bytes): Undefined and MUST be ignored.

recipientCount (4 bytes): An unsigned integer that specifies the count of strings in the collection of recipients specified by **rgRecipientRoutingSlipStrings**.

currentRecipient (4 bytes): An unsigned integer that specifies the addressee of the document routing slip. It MUST be less than or equal to **recipientCount**+1. A value of 0x00000000 or a value of **recipientCount**+1 specifies the originator identified by **originatorString**. A value greater than 0x00000000 and less than **recipientCount**+1 specifies a 1-based index into the collection of recipients specified by **rgRecipientRoutingSlipStrings**.

A - fOneAfterAnother (1 bit): A bit that specifies how a document is sent to recipients. It MUST be a value from the following table.

Value	Meaning
FALSE	The document is sent from the originator to all

Value	Meaning
	recipients simultaneously.
TRUE	The document is sent sequentially to one recipient after another. After a recipient is finished processing the document, the document is sent to the next recipient.

B - fReturnWhenDone (1 bit): A bit that specifies whether a document is sent back to the originator after all recipients have processed the document.

C - fTrackStatus (1 bit): A bit that specifies whether progress of a document routing slip is tracked. If progress is tracked, the originator is notified after a recipient finishes processing the document.

D - reserved1 (1 bit): MUST be zero, and MUST be ignored.

E - fDocumentRouted (1 bit): A bit that specifies whether the document-routing is in progress and the document is being processed by recipients.

F - fCycleCompleted (1 bit): A bit that specifies whether all recipients have finished processing the document.

reserved2 (26 bits): MUST be ignored and MUST be 0.

unused2 (4 bytes): Undefined and MUST be ignored.

originatorString (variable): A DocRoutingSlipString structure that specifies the originator of a document routing slip. The **originatorString.stringType** field MUST be 0x0001.

rgRecipientRoutingSlipStrings (variable): An array of DocRoutingSlipString structures that specifies recipients of a document routing slip. The count of items in the array is specified by **recipientCount**. The **stringType** field of each DocRoutingSlipString item MUST be 0x0002.

subjectString (variable): A DocRoutingSlipString structure that specifies the subject of a document routing slip. The **subjectString.stringType** field MUST be 0x0003.

messageString (variable): A DocRoutingSlipString structure that specifies the message of a document routing slip. The **messageString.stringType** field MUST be 0x0004.

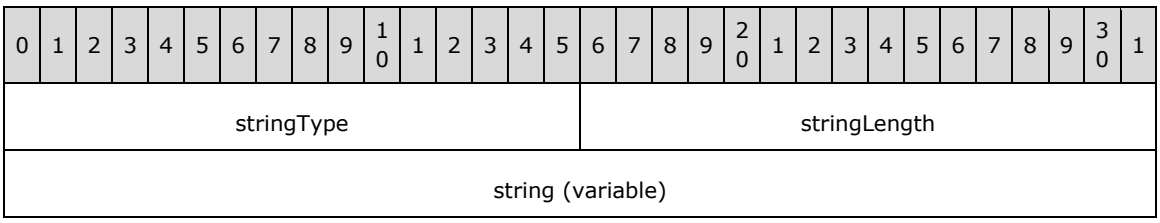
unused3 (variable): Undefined and MUST be ignored. The length, in bytes, of the field is specified by the following formula:

```
8 + rh.recLen – length
```

2.11.2 DocRoutingSlipString

Referenced by: DocRoutingSlipAtom

A structure that specifies information about a string in a document routing slip.



...

stringType (2 bytes): An **unsigned integer** that specifies the type of a string. It MUST be a value from the following table.

Value	Meaning
0x0001	The originator of a document routing slip.
0x0002	A recipient of a document routing slip.
0x0003	The subject of a document routing slip.
0x0004	The message body of a document routing slip.

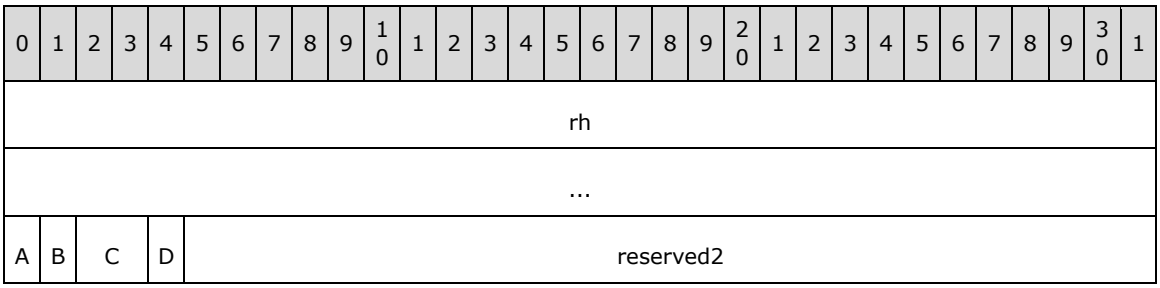
stringLength (2 bytes): An unsigned integer that specifies the number of bytes supplied for the string, minus 1. If **stringType** is equal to 0x0001 or 0x0002, this value MUST be greater than 0x0000.

string (variable): A **PrintableAnsiString** (section 2.2.22) that specifies the characters of the string. The length, in bytes, of the string is specified by **stringLength** plus 1. If **stringType** is equal to 0x0001 or 0x0002, the byte at index **stringLength** minus 1 MUST be equal to 0x00 and the byte at index **stringLength** MUST be ignored. If **stringType** is equal to 0x0003 or 0x0004, the byte at index **stringLength** MUST be equal to 0x00.

2.11.3 EnvelopeFlags9Atom

Referenced by: PP9DocBinaryTagExtension

An atom record that specifies information about an envelope.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_EnvelopeFlags9Atom.
rh.recLen	MUST be 0x00000004.

A - fHasEnvelope (1 bit): A bit that specifies whether an EnvelopeData2.11.4Atom record exists in the file.

B - fEnvelopeVisible (1 bit): A bit that specifies whether the envelope is visible. If the value is **TRUE**, **fHasEnvelope** MUST also be **TRUE**.

C - reserved1 (2 bits): MUST be zero and MUST be ignored.

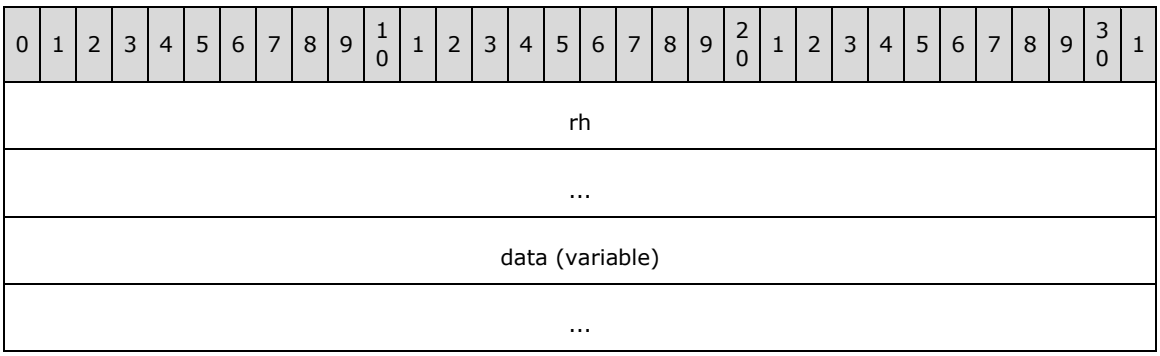
D - fEnvelopeDirty (1 bit): A bit that specifies whether the envelope has been modified since the last time it was sent to the mail client. If the value is **TRUE**, **fHasEnvelope** MUST also be **TRUE**.

reserved2 (27 bits): MUST be zero and MUST be ignored.

2.11.4 EnvelopeData9Atom

Referenced by: PP9DocBinaryTagExtension

An atom record that specifies data for an envelope.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

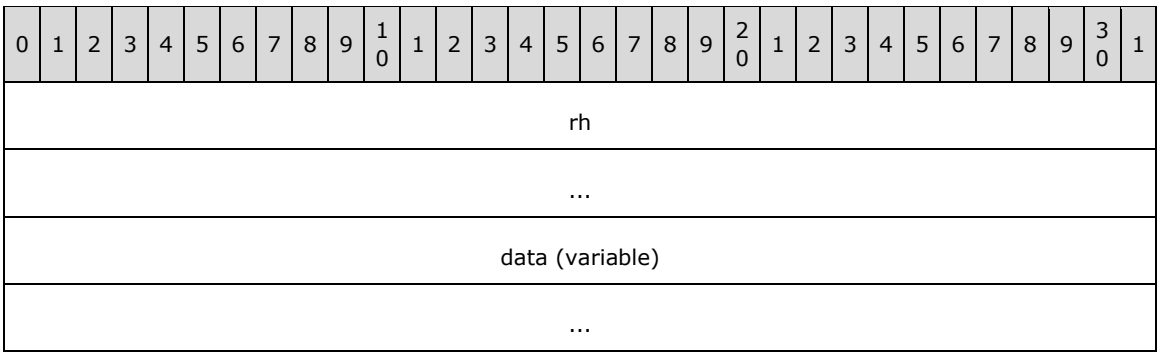
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_EnvelopeData9Atom.

data (variable): An **MsoEnvelopeCLSID** ([MS-OSHARED] section 2.3.8.1) that specifies data for an envelope. The length, in bytes, of this field is specified by **rh.recLen**.

2.11.5 FontEmbedDataBlob

Referenced by: FontCollectionEntry

An atom record that specifies the font data of an embedded font.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

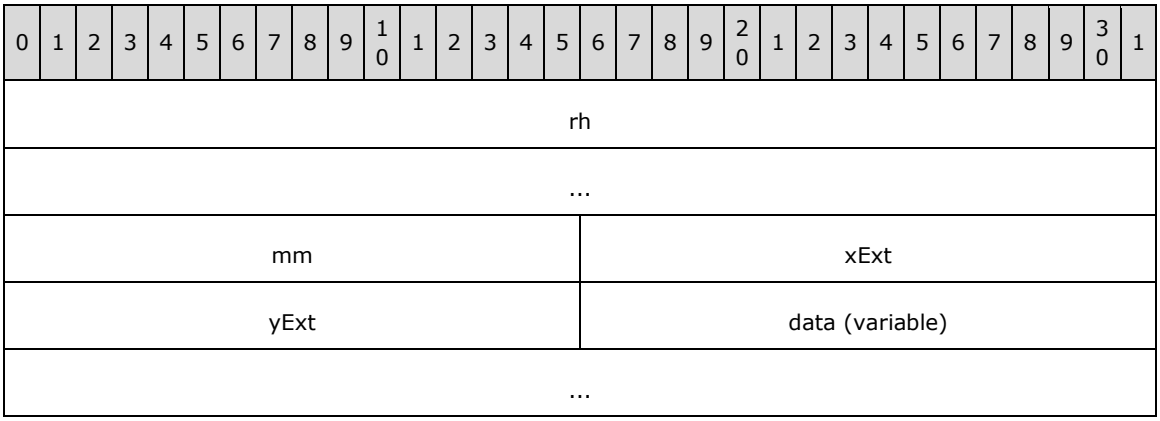
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be greater than or equal to 0x000 and less than or equal to 0x003.
rh.recType	MUST be an RT_FontEmbedDataBlob.

data (variable): A structure that specifies the font data of an embedded font as specified in [Embed-Open-Type-Format]. The length, in bytes, of this field is specified by **rh.recLen**.

2.11.6 MetafileBlob

Referenced by: ExControlContainer, ExOleEmbedContainer, ExOleLinkContainer

An atom record that specifies a metafile ([MS-WMF]).



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_MetaFile.
rh.recLen	MUST be greater than 0x00000010.

mm (2 bytes): A signed integer that specifies the mapping mode of the metafile as specified in [MS-WMF] section 2.1.1.16.

xExt (2 bytes): A signed integer that specifies the width of the metafile in units that correspond to the mapping mode specified by the **mm** field as specified in [MS-WMF] section 2.1.1.16.

yExt (2 bytes): A signed integer that specifies the height of the metafile in units that correspond to the mapping mode specified by the **mm** field as specified in [MS-WMF] section 2.1.1.16.

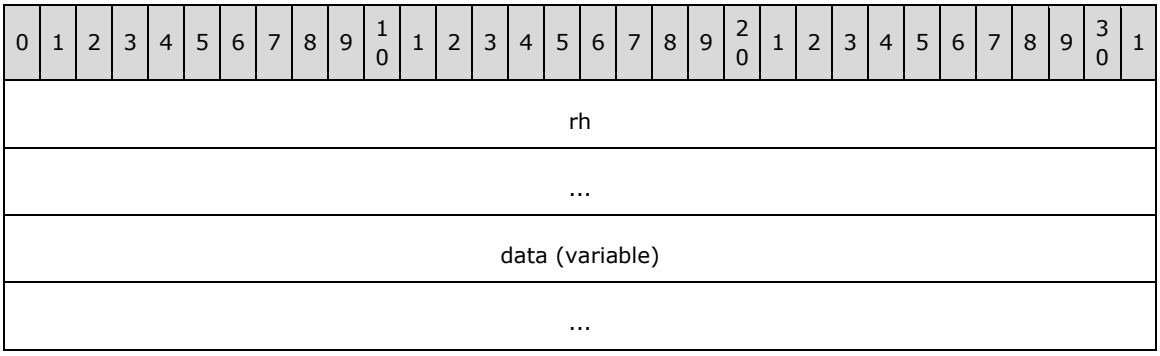
data (variable): A metafile as specified in [MS-WMF]. The length, in bytes, of the field is specified by the following formula:

`rh.recLen - 6.`

2.11.7 RoundTripAnimationAtom

Referenced by: RoundTripMainMasterRecord, RoundTripSlideRecord

An atom record that specifies animations for a slide.



rh (8 bytes): A RecordHeader type that specifies the header for this record. Sub-fields are further specified in the following table.

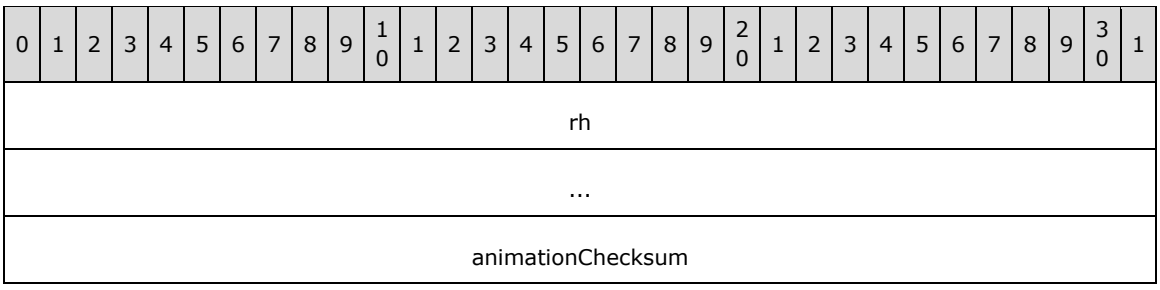
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_RoundTripAnimationAtom12Atom.

data (variable): An ECMA-376 document that specifies animations, along with embedded sounds if present. The package contains XML in the PresentationML Timing Info part containing a <timing> element that conforms to the schema specified by **CT_SlideTiming** as specified in [ECMA-376] Part 4: Markup Language Reference, section 4.4.1.44.

2.11.8 RoundTripAnimationHashAtom

Referenced by: RoundTripMainMasterRecord, RoundTripSlideRecord

An atom record that specifies a checksum for animation data.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_RoundTripAnimationHashAtom12Atom.
rh.recLen	MUST be 0x00000004.

animationChecksum (4 bytes): An **unsigned integer** that specifies the checksum of the animation data.

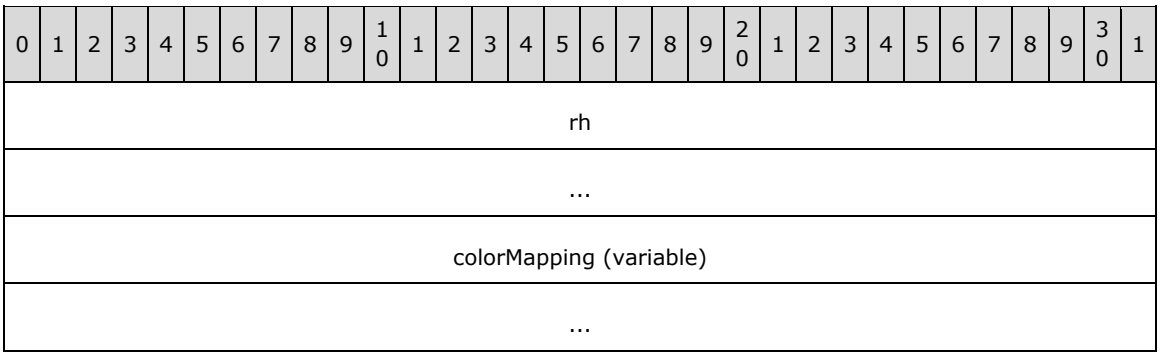
Let *corresponding slide* be specified as the **SlideContainer** record (section 2.5.1) that contains this **RoundTripAnimationHashAtom** record.

The data used to calculate the checksum is all fields in the VisualSoundAtom record contained within the *corresponding slide*, computed sequentially in 4-byte pieces. The other input to the checksum calculation is all of the bytes of the **spid** field of the **OfficeArtFSP** record ([MS-ODRAW] section 2.2.40) that specify the shape identifier of each shape on the corresponding slide. The checksum value is a **cyclic redundancy check (CRC)** logical exclusive or (XOR) hash of each consecutive 4-byte sequence in the specified data.

2.11.9 RoundTripColorMappingAtom

Referenced by: HandoutRoundTripAtom, NotesRoundTripAtom, RoundTripMainMasterRecord, RoundTripSlideRecord

An atom record that specifies the color mapping for a slide.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

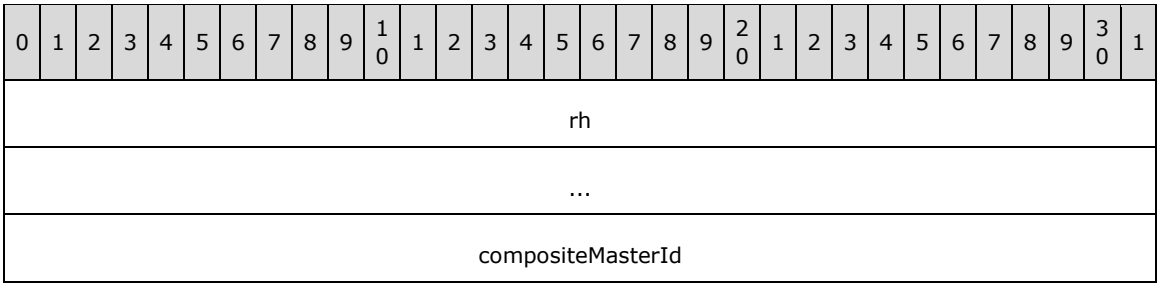
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_RoundTripColorMapping12Atom.

colorMapping (variable): A Utf2.2.36UnicodeString that specifies the color mapping. Either the XML in this string contains a **clrMap** element that conforms to the schema specified by **CT_ColorMapping** as specified in [ECMA-376] Part 4: Markup Language Reference, section 4.4.1.6, or the XML contains a **clrMapOverride** element that conforms to the schema specified by **CT_ColorMappingOverride** as specified in [ECMA-376] Part 4: Markup Language Reference, section 4.4.1.7.

2.11.10 RoundTripCompositeMasterId12Atom

Referenced by: RoundTripMainMasterRecord, RoundTripSlideRecord

An atom record that specifies that a slide is a slide layout merged with its main master slide. The *corresponding main master slide* has a RoundTripOriginalMainMasterId2.11.20Atom record with the same identifier.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

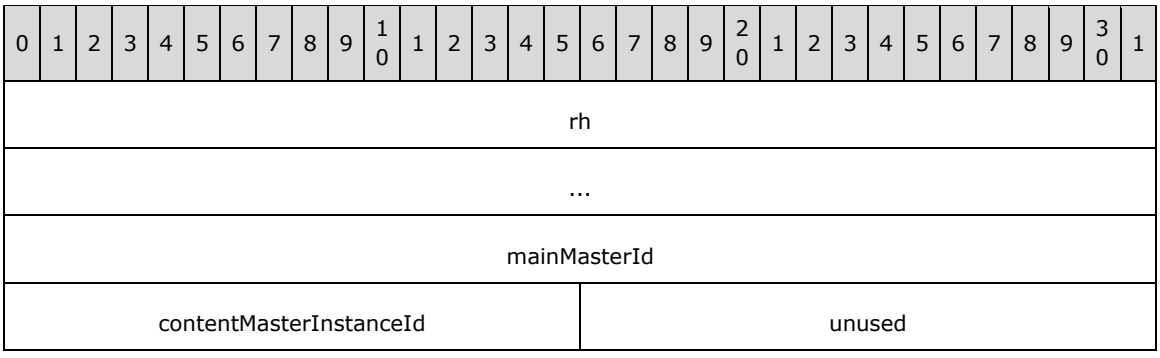
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_RoundTripCompositeMasterId12Atom.
rh.recLen	MUST be 0x00000004.

compositeMasterId (4 bytes): An **unsigned integer** that specifies the identifier for the main master slide this slide was merged with. This field is equivalent to the **ST_SlideMasterId** simple type as specified in [ECMA-376] Part 4: Markup Language Reference, section 4.8.20.

2.11.11 RoundTripContentMasterId12Atom

Referenced by: RoundTripSlideRecord

An atom record that specifies the relation between a slide and its slide layout.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be an RT_RoundTripContentMasterId12Atom.
rh.recLen	MUST be 0x00000008.

mainMasterId (4 bytes): An unsigned integer that specifies the identifier of a main master slide. This field is equivalent to the **ST_SlideMasterId** simple type as specified in [ECMA-376] Part 4: Markup Language Reference, section 4.8.20.

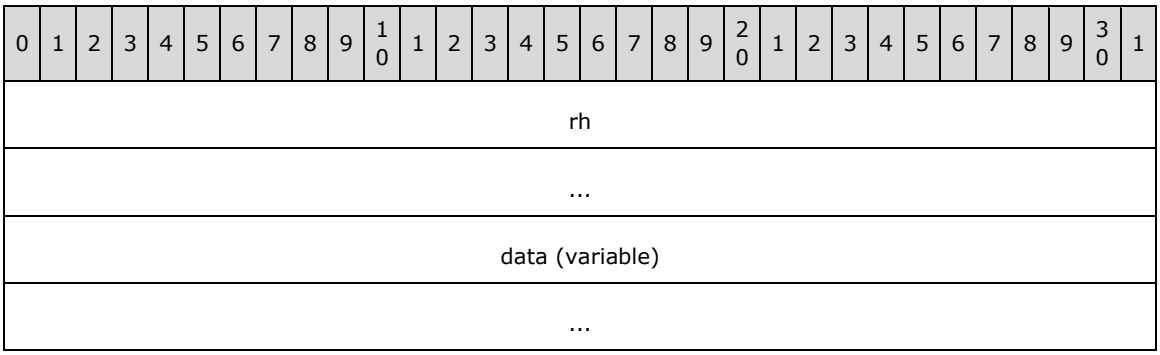
contentMasterInstanceId (2 bytes): An unsigned integer that specifies the instance identifier of the slide layout. This field is equivalent to the **ST_SlideLayoutId** simple type as specified in [ECMA-376] Part 4: Markup Language Reference, section 4.8.18.

unused (2 bytes): Undefined and MUST be ignored.

2.11.12 RoundTripContentMasterInfo12Atom

Referenced by: RoundTripMainMasterRecord

An atom record that specifies a slide layout.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

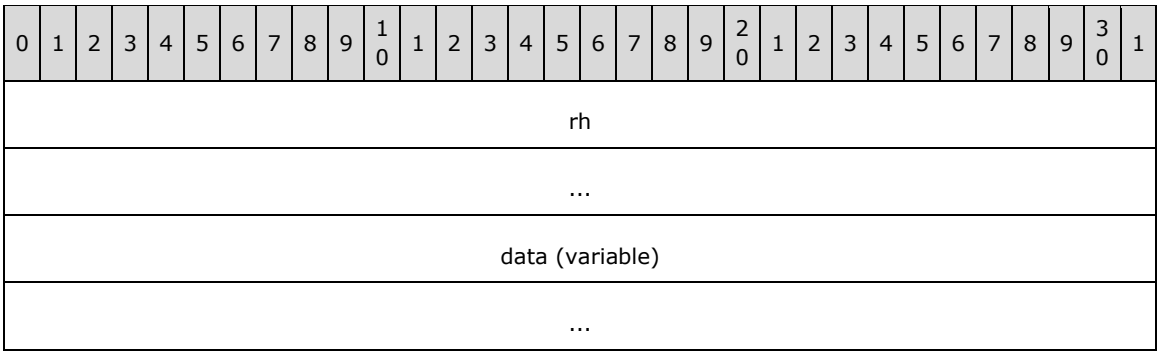
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recType	MUST be an RT_RoundTripContentMasterInfo12Atom.

data (variable): An ECMA-376 document that specifies a slide layout. The package contains XML in the PresentationML Content Master part containing a **sldLayout** element that conforms to the schema specified by **CT_SlideLayout** as specified in [ECMA-376] Part 4: Markup Language Reference, section 4.4.1.36.

2.11.13 RoundTripCustomTableStyles12Atom

Referenced by: DocumentContainer

An atom record that specifies table styles.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

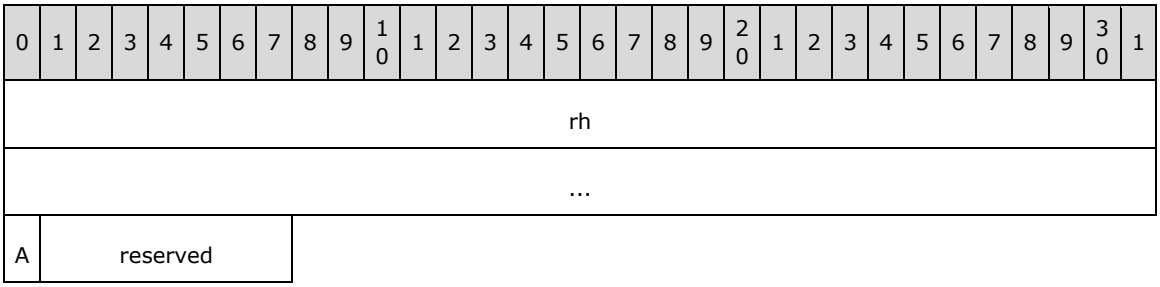
Field	Meaning
rh.recVer	MUST<110> be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_RoundTripCustomTableStyles12Atom.

data (variable): An ECMA-376 document that specifies table styles. The package contains XML in the PresentationML Table Styles part containing a **tblStyle** element that conforms to the schema specified by **CT_TableStyleList** as specified in [ECMA-376] Part 4: Markup Language Reference, section 5.1.4.2.27.

2.11.14 RoundTripDocFlags12Atom

Referenced by: PP12DocBinaryTagExtension

An atom record that specifies document-level flags.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_RoundTripDocFlags12Atom.
rh.recLen	MUST be 0x00000001.

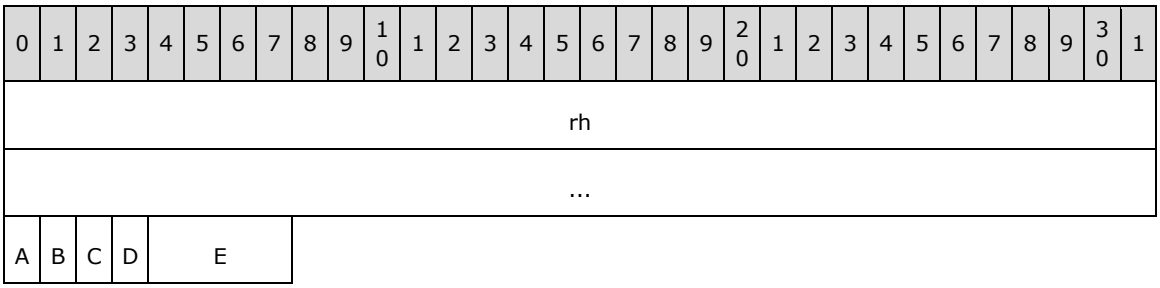
A - fCompressPicturesOnSave (1 bit): A bit that specifies whether pictures are automatically compressed when saving.

reserved (7 bits): MUST be zero and MUST be ignored.

2.11.15 RoundTripHeaderFooterDefaults12Atom

Referenced by: PP12SlideBinaryTagExtension

An atom record that specifies default header and footer flags.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_RoundTripHeaderFooterDefaults12Atom.
rh.recLen	MUST be 0x00000001.

A - fIncludeDate (1 bit): A bit that specifies whether the date is included in the footer of new slides.

B - fIncludeFooter (1 bit): A bit that specifies whether the footer is included in new slides.

C - fIncludeHeader (1 bit): A bit that specifies whether the header is included in new slides.

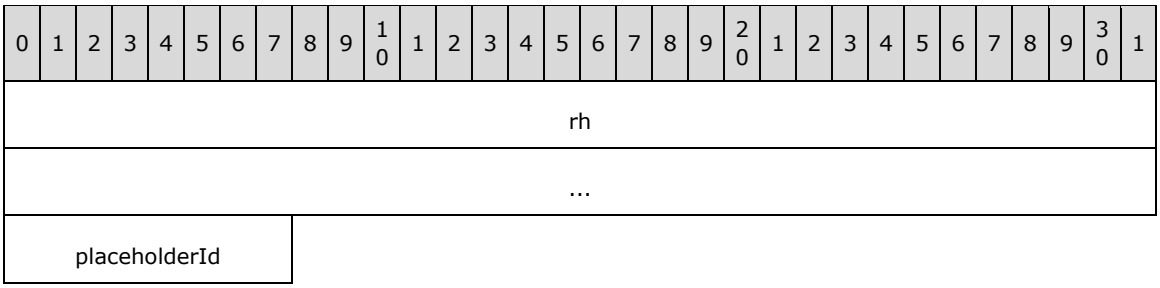
D - fIncludeSlideNumber (1 bit): A bit that specifies whether the slide number or page number is included in the footer of new slides.

E - reserved (4 bits): MUST be zero and MUST be ignored.

2.11.16 RoundTripHFPlaceholder12Atom

Referenced by: ShapeClientRoundtripDataSubContainerOrAtom

An atom record that specifies that a shape is a header or footer placeholder shape.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

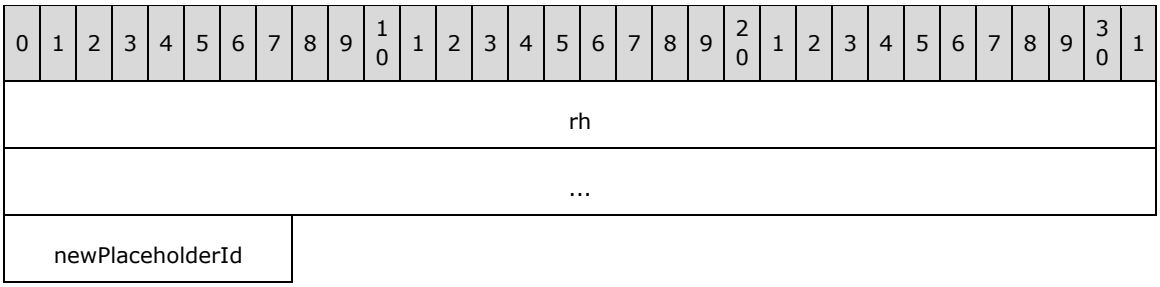
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_RoundTripHFPlaceholder12Atom.
rh.recLen	MUST be 0x00000001.

placeholderId (1 byte): A **PlaceholderEnum** enumeration that specifies the placeholder shape identifier. It MUST be PT_MasterDate, PT_MasterSlideNumber, PT_MasterFooter, or PT_MasterHeader.

2.11.17 RoundTripNewPlaceholderId12Atom

Referenced by: ShapeClientRoundtripDataSubContainerOrAtom

An atom record that specifies a **placeholder** identifier.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

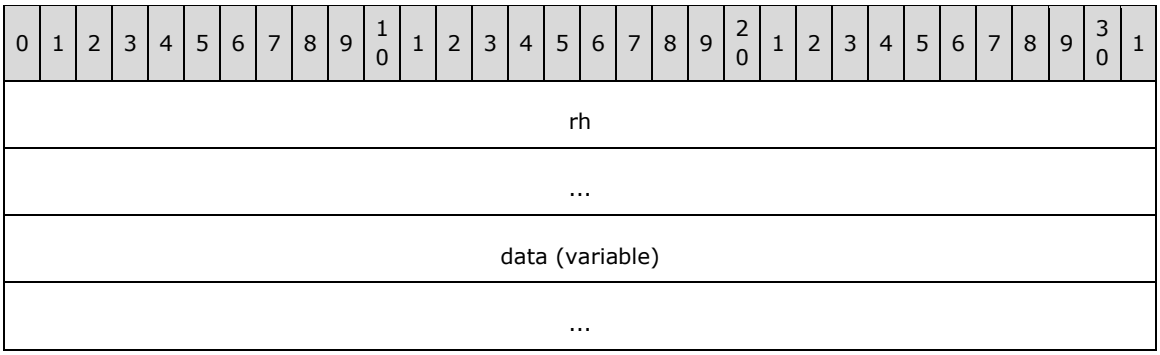
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_RoundTripNewPlaceholderId12Atom.
rh.recLen	MUST be 0x00000001.

newPlaceholderId (1 byte): A **PlaceholderEnum** enumeration that specifies the placeholder shape identifier. It MUST be PT_VerticalObject or PT_Picture.

2.11.18 RoundTripNotesMasterTextStyles12Atom

Referenced by: NotesRoundTripAtom

An atom record that specifies text styles used by the notes master slide.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

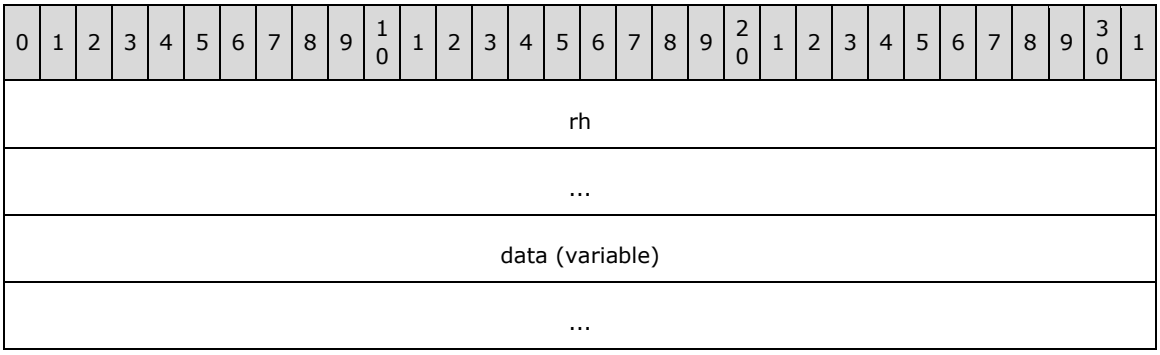
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_RoundTripNotesMasterTextStyles12Atom.

data (variable): An ECMA-376 document that specifies text styles. The package contains XML in the PresentationML Main Master part containing a **txStyles** element that conforms to the schema specified by **CT_SlideMasterTextStyles** as specified in [ECMA-376] Part 4: Markup Language Reference, section 4.4.1.48.

2.11.19 RoundTripOArtTextStyles12Atom

Referenced by: MainMasterContainer, RoundTripMainMasterRecord

An atom record that specifies text styles used by a main master slide.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

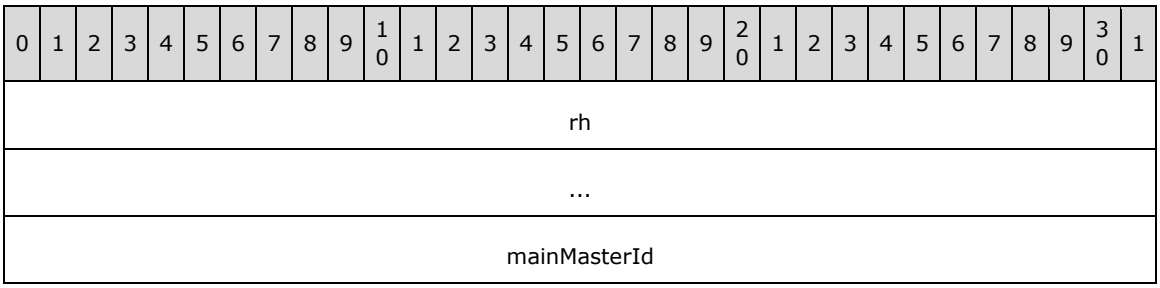
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_RoundTripOArtTextStyles12Atom.

data (variable): An ECMA-376 document that specifies text styles. The package contains XML in the PresentationML Main Master part containing a **txStyles** element that conforms to the schema specified by **CT_SlideMasterTextStyles** as specified in [ECMA-376] Part 4: Markup Language Reference, section 4.4.1.48.

2.11.20 RoundTripOriginalMainMasterId12Atom

Referenced by: RoundTripMainMasterRecord

An atom record that specifies the original identifier of a <sldMaster> element in the Slide Master part that conforms to the schema specified by **CT_SlideMaster** as specified in [ECMA-376] Part 4: Markup Language Reference, section 4.4.1.39.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

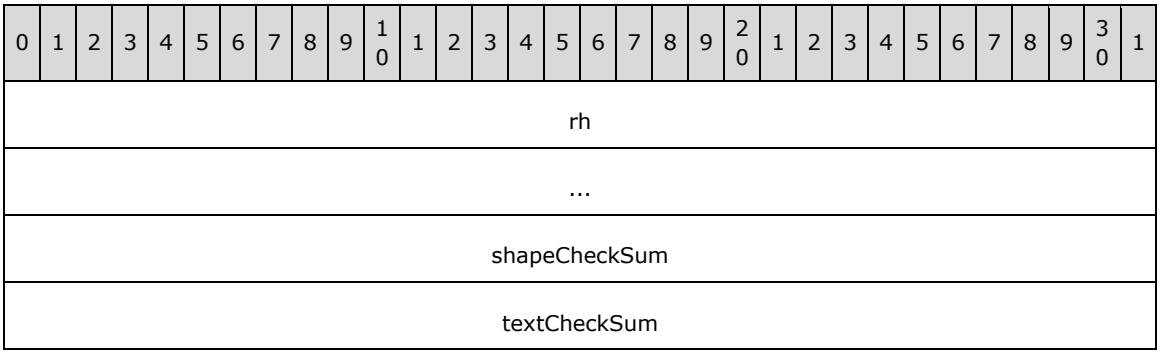
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_RoundTripOriginalMainMasterId12Atom.
rh.recLen	MUST be 0x00000004.

mainMasterId (4 bytes): An **unsigned integer** that specifies the identifier of the main master slide. This field is equivalent to the **ST_SlideMasterId** simple type as specified in [ECMA-376] Part 4: Markup Language Reference, section 4.8.20.

2.11.21 RoundTripShapeCheckSumForCustomLayouts12Atom

Referenced by: ShapeClientRoundtripDataSubContainerOrAtom

An atom record that specifies application-defined IDs for a shape and its text. To be interoperable this record SHOULD<111> be preserved if encountered but SHOULD<112> be omitted otherwise.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_RoundTripShapeCheckSumForCL12Atom.
rh.recLen	MUST be 0x00000008.

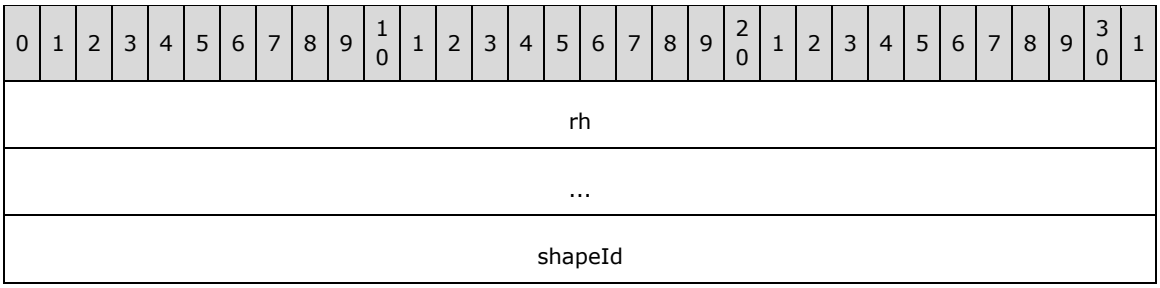
shapeCheckSum (4 bytes): An unsigned integer that specifies an application-defined identifier for quickly determining whether the shape properties specified by the **OfficeArtSpContainer** record ([MS-ODRAW] section 2.2.14) that contains this **RoundTripShapeChecksumForCustomLayouts12Atom** have changed since they were last saved.

textCheckSum (4 bytes): An unsigned integer that specifies an application-defined identifier for quickly determining whether the text body specified by the OfficeArtClientTextbox record contained within the **OfficeArtSpContainer** record ([MS-ODRAW] section 2.2.14) that contains this **RoundTripShapeChecksumForCustomLayouts12Atom** has changed since it was last saved.

2.11.22 RoundTripShapeId12Atom

Referenced by: ShapeClientRoundtripDataSubContainerOrAtom

An atom record that specifies a shape identifier.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

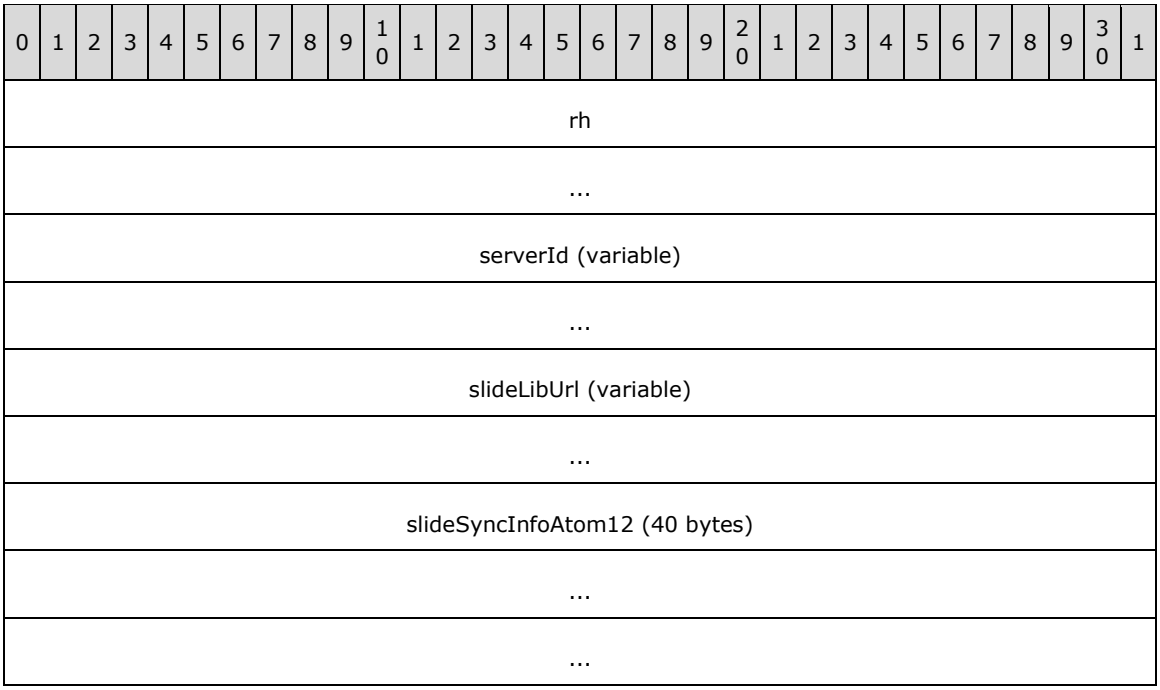
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_RoundTripShapeId12Atom.
rh.recLen	MUST be 0x00000004.

shapeId (4 bytes): An unsigned integer that specifies the shape identifier. This field is equivalent to the **ST_ShapeID** simple type as specified in [ECMA-376] Part 4: Markup Language Reference, section 5.1.12.55.

2.11.23 RoundTripSlideSyncInfo12Container

Referenced by: RoundTripSlideRecord, SlideContainer

A container record that specifies information about a slide that synchronizes to a slide in a slide library. Slide synchronization data is fully specified in [ECMA-376] Part 4: Markup Language Reference, section 4.7.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_RoundTripSlideSyncInfo12.

serverId (variable): A ServerIdAtom record that specifies a unique identifier for a slide in a slide library.

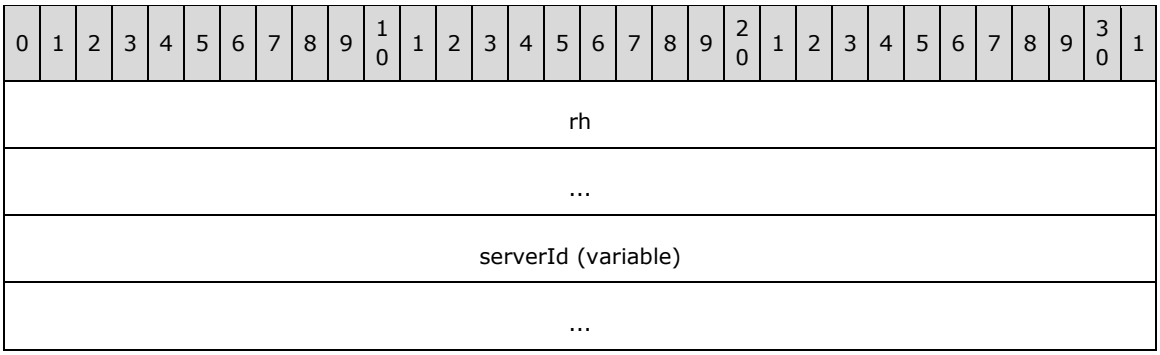
slideLibUrl (variable): A SlideLibUrlAtom record that specifies the URL of a slide library.

slideSyncInfoAtom12 (40 bytes): A SlideSyncInfoAtom2.11.26 record that specifies timestamps for slides that synchronize with versions stored in a slide library.

2.11.24 ServerIdAtom

Referenced by: RoundTripSlideSyncInfo12Container

An atom record that specifies a unique identifier for a slide in a slide library.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

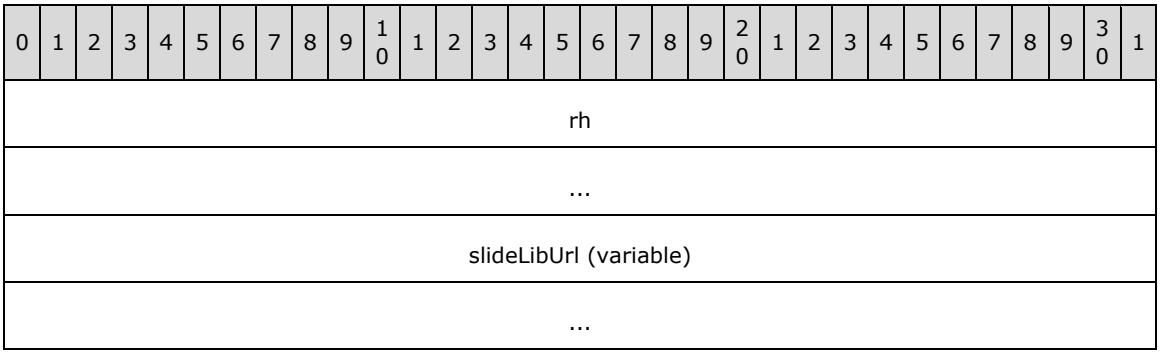
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an **RT_CString** (section 2.13.24).
rh.recLen	MUST be even.

serverId (variable): A **PrintableUnicodeString** (section 2.2.23) that specifies the unique identifier for the slide. This field is equivalent to the **serverSldId** attribute specified in [ECMA-376] Part 4: Markup Language Reference, section 4.7.1.

2.11.25 SlideLibUrlAtom

Referenced by: RoundTripSlideSyncInfo12Container

An atom record that specifies the URL of a slide library.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

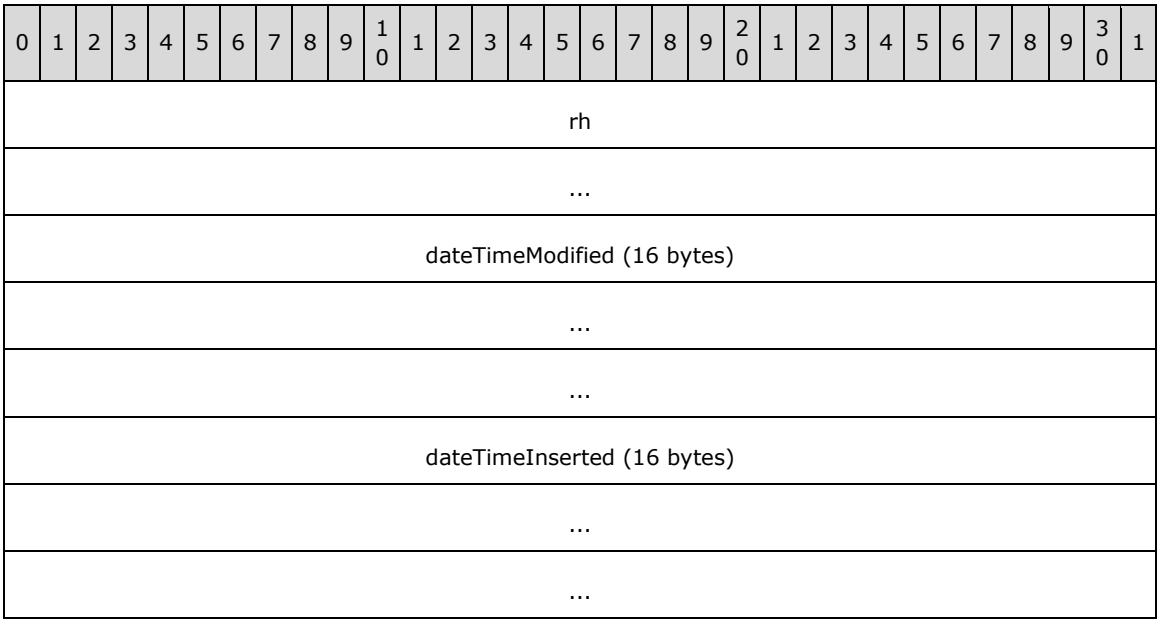
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be an **RT_CString** (section 2.13.24).
rh.recLen	MUST be even.

slideLibUrl (variable): An HttpUrl that specifies the URL of the slide library.

2.11.26 SlideSyncInfoAtom12

Referenced by: RoundTripSlideSyncInfo12Container

An atom record that specifies timestamps for a slide that synchronizes with a version of the slide stored in a slide library.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_RoundTripSlideSyncInfoAtom12.
rh.recLen	MUST be 0x00000020.

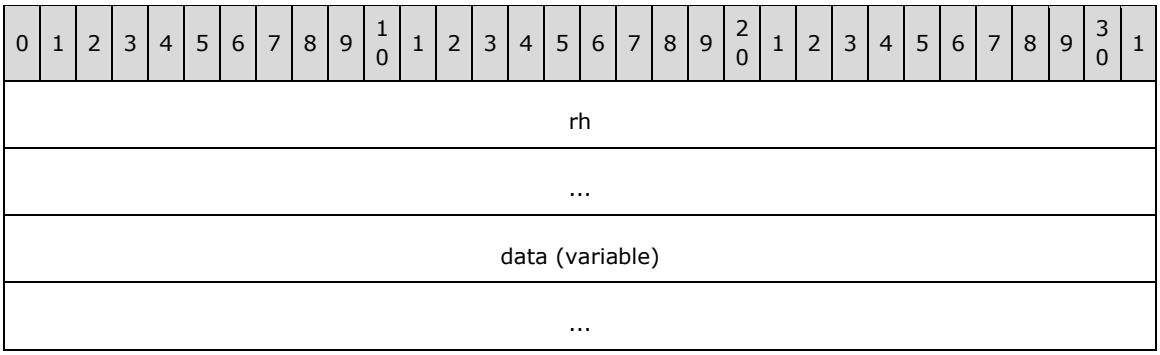
dateTimeModified (16 bytes): A DateTimeStruct structure that specifies the time stamp the slide was last modified on the server. This field is equivalent to the **serverSldModified** attribute as specified in [ECMA-376] Part 4: Markup Language Reference, section 4.7.1.

dateTimeInserted (16 bytes): A DateTimeStruct structure that specifies the time stamp the slide was inserted in the document. This field is equivalent to the **clientInsertedTime** attribute as specified in [ECMA-376] Part 4: Markup Language Reference, section 4.7.1.

2.11.27 RoundTripThemeAtom

Referenced by: HandoutRoundTripAtom, NotesRoundTripAtom, RoundTripMainMasterRecord, RoundTripSlideRecord

An atom record that specifies the theme of the main master slide.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

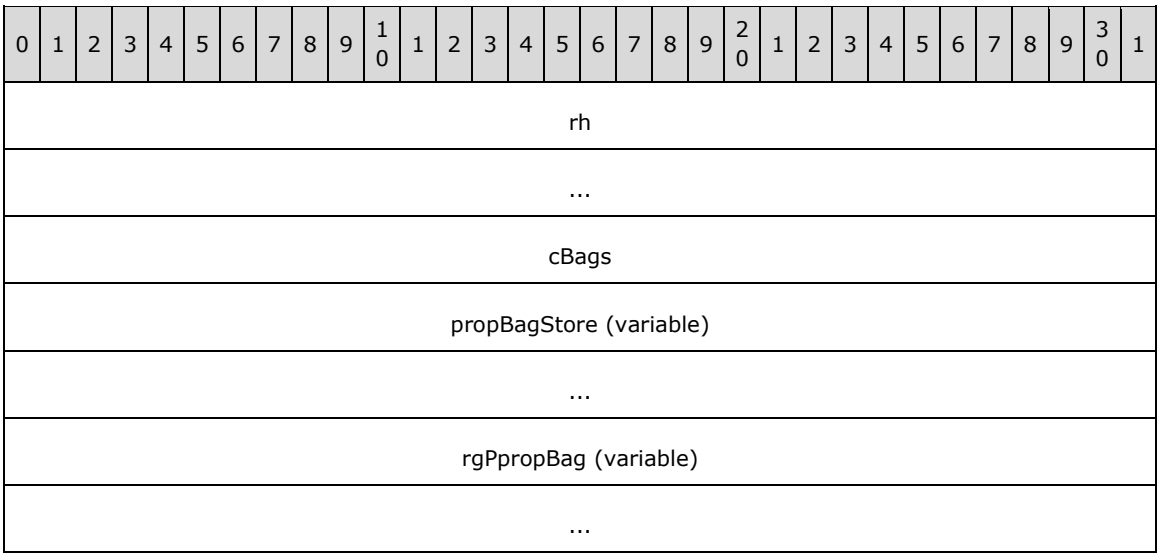
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_RoundTripTheme12Atom.

data (variable): An ECMA-376 document that specifies theme information. The package contains XML in the DrawingML Theme part containing a **theme** element that conforms to the schema specified by **CT_OfficeStyleSheet** as specified in [ECMA-376] Part 4: Markup Language Reference, section 5.1.8.9, or XML in the DrawingML Theme Override part containing a **themeOverride** element that conforms to the schema specified by **CT_BaseStylesOverride** as specified in [ECMA-376] Part 4: Markup Language Reference, section 5.1.8.12.

2.11.28 SmartTagStore11Container

Referenced by: PP11DocBinaryTagExtension

A container record that specifies information about all smart tags within the corresponding presentation. A smart tag is additional information that is specified to correspond to a string of text.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_SmartTagStore11Container.

cBags (4 bytes): An unsigned integer that specifies the count of items in **rgPpropBag.**

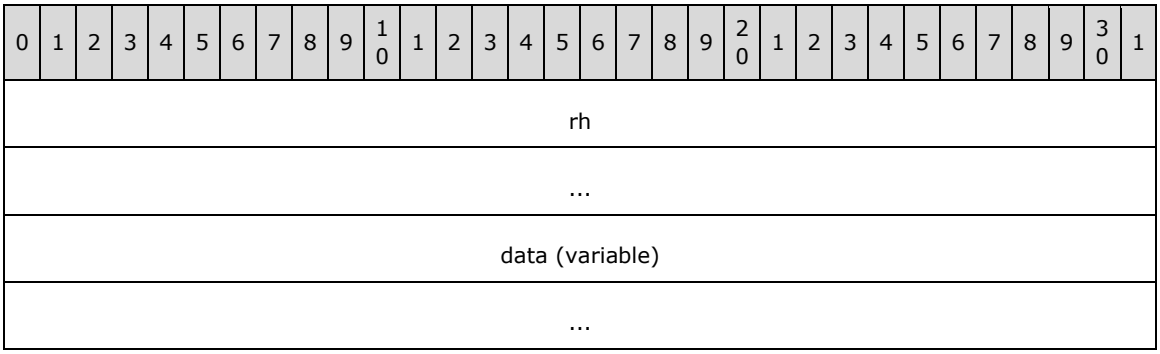
propBagStore (variable): A **PropertyBagStore** that specifies a collection of smart tag types and their corresponding data as specified in [MS-OSHARED] section 2.3.4.1.

rgPpropBag (variable): An array of **PropertyBag** that specifies a set of properties with their corresponding key/value pairs as specified in [MS-OSHARED] section 2.3.4.4. These key/value pairs each represent a string of text and correspond to an entry within **propBagStore**.

2.11.29 SoundDataBlob

Referenced by: SoundContainer

An atom record that specifies audio data for a sound.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_SoundDataBlob.

data (variable): A structure that specifies WAV or AIFF audio data for a sound. The length, in bytes, of this field is specified by **rh.recLen**.

2.11.30 ProgStringTagContainer

Referenced by: DocProgTagsSubContainerOrAtom, ShapeProgTagsSubContainerOrAtom, SlideProgTagsSubContainerOrAtom

A container record that specifies a programmable tag that has a UnicodeString as its value.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
tagNameAtom (variable)																															
...																															
tagValueAtom (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0xF.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_ProgStringTag.

tagNameAtom (variable): A TagNameAtom record that specifies the name of the programmable tag.

tagValueAtom (variable): An optional TagValueAtom record that specifies the value of the programmable tag.

2.11.31 TagNameAtom

Referenced by: ProgStringTagContainer, UnknownBinaryTag

An atom record that contains the name of the name-value pair in a programmable tag.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
tagName (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.

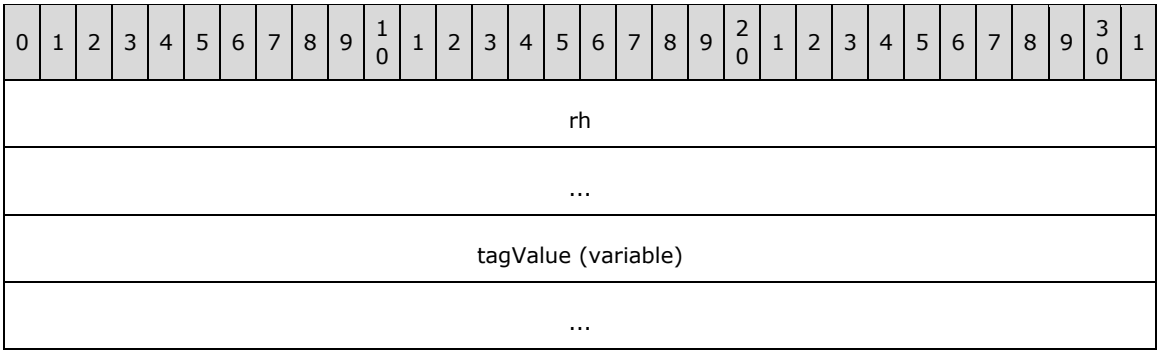
Field	Meaning
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be greater than zero.

tagName (variable): A **PrintableUnicodeString** string (section 2.2.23) that specifies the name of the programmable tag. The length, in bytes, of the field is specified by **rh.recLen**.

2.11.32 TagValueAtom

Referenced by: ProgStringTagContainer

An atom record that contains the value of the name-value pair in a programmable tag.



rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

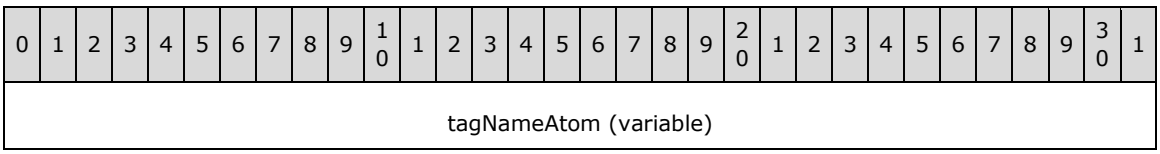
Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x001.
rh.recType	MUST be an **RT_CString** (section 2.13.24).
rh.recLen	MUST be an even number. It MUST be greater than zero.

tagValue (variable): A UnicodeString string that specifies the value of the programmable tag. The length, in bytes, of the field is specified by **rh.recLen**.

2.11.33 UnknownBinaryTag

Referenced by: DocProgBinaryTagSubContainerOrAtom, ShapeProgBinaryTagSubContainerOrAtom, SlideProgBinaryTagSubContainerOrAtom

A pair of atom records that specifies a programmable tag that has binary data as its value.



...
tagData (variable)
...

tagNameAtom (variable): A TagNameAtom record that specifies the name of the programmable tag.

tagData (variable): A BinaryTagDataBlob record that specifies the value of the programmable tag.

2.11.34 BinaryTagDataBlob

Referenced by: UnknownBinaryTag

An atom record that contains the value of the name-value pair in a programmable tag.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
rh																															
...																															
data (variable)																															
...																															

rh (8 bytes): A **RecordHeader** structure (section 2.3.1) that specifies the header for this record. Sub-fields are further specified in the following table.

Field	Meaning
rh.recVer	MUST be 0x0.
rh.recInstance	MUST be 0x000.
rh.recType	MUST be an RT_BinaryTagDataBlob.

data (variable): An array of bytes that specifies the data of this item. The size, in bytes, of the data is specified by **rh.recLen**.

2.12 Common Structures

2.12.1 ColorStruct

Referenced by: SchemeListElementColorSchemeAtom, SlideSchemeColorSchemeAtom

A structure that specifies a color in the sRGB color space as specified in [IEC-RGB].

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
red								green								blue								unused							

red (1 byte): An unsigned integer that specifies the red component of this color.

green (1 byte): An unsigned integer that specifies the green component of this color.

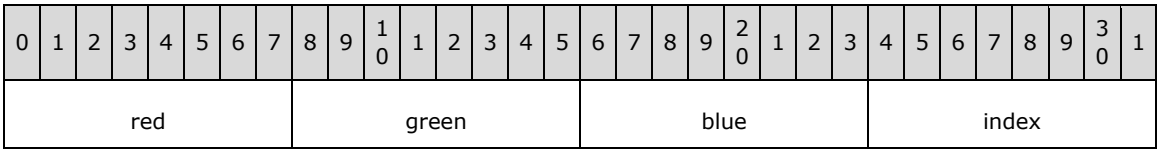
blue (1 byte): An unsigned integer that specifies the blue component of this color.

unused (1 byte): Undefined and MUST be ignored.

2.12.2 ColorIndexStruct

Referenced by: AnimationInfoAtom, SlideShowDocInfoAtom, TextCFException, TextPFException

A structure that specifies an index in the color scheme, or a color in the sRGB color space as specified in [IEC-RGB]. Color schemes are specified by the SlideSchemeColorSchemeAtom record.



red (1 byte): An unsigned integer that specifies the red component of this color.

green (1 byte): An unsigned integer that specifies the green component of this color.

blue (1 byte): An unsigned integer that specifies the blue component of this color.

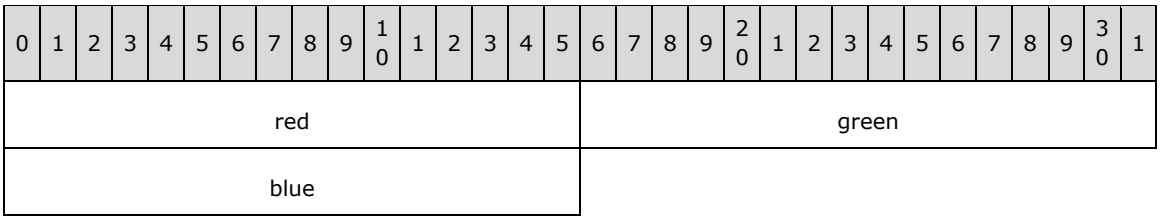
index (1 byte): An unsigned integer that specifies the index in the color scheme. It MUST be a value from the following table.

Value	Description
0x00	Background color
0x01	Text color
0x02	Shadow color
0x03	Title text color
0x04	Fill color
0x05	Accent 1 color
0x06	Accent 2 color
0x07	Accent 3 color
0xFE	Color is an sRGB value specified by **red**, **green**, and **blue** fields.
0xFF	Color is undefined.

2.12.3 WideColorStruct

Referenced by: RecolorEntry, RecolorEntryBrush, RecolorEntryColor, RecolorInfoAtom

A structure that specifies a color in the sRGB color space as specified in [IEC-RGB].



red (2 bytes): An unsigned integer that specifies the red component of this color.

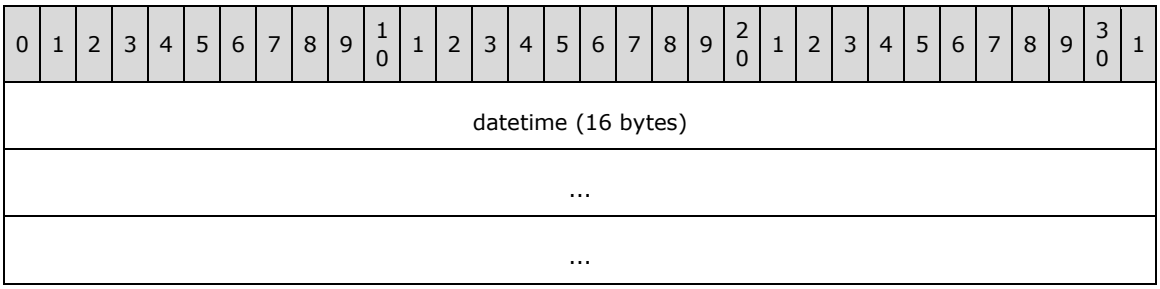
green (2 bytes): An unsigned integer that specifies the green component of this color.

blue (2 bytes): An unsigned integer that specifies the blue component of this color.

2.12.4 DateTimeStruct

Referenced by: BroadcastDocInfoAtom, Comment10Atom, SlideSyncInfoAtom12

A structure that specifies the date and time.

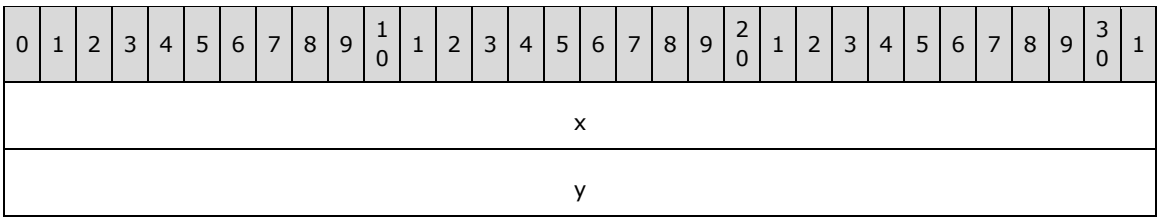


datetime (16 bytes): A **SYSTEMTIME** structure, as specified in [MS-DTYP] section 2.3.13, that specifies the date and time.

2.12.5 PointStruct

Referenced by: Comment10Atom, DocumentAtom, NoZoomViewInfoAtom, ZoomViewInfoAtom

A structure that specifies a point in the *x-y* coordinate system.



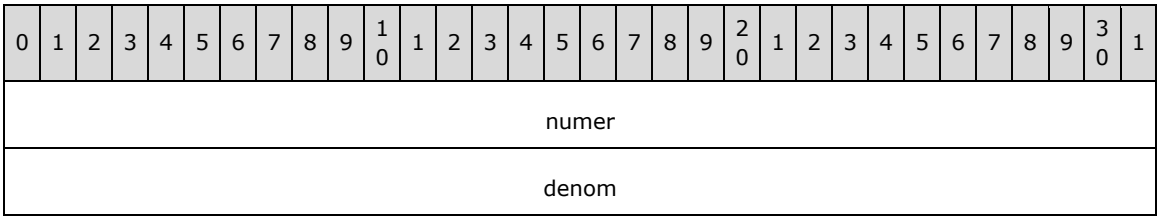
x (4 bytes): A signed integer that specifies the *x*-coordinate. Positive *x* increases to the right.

y (4 bytes): A signed integer that specifies the *y*-coordinate. Positive *y* increases to the bottom.

2.12.6 RatioStruct

Referenced by: DocumentAtom, NormalViewSetInfoAtom, ScalingStruct

A structure that specifies a rational number.



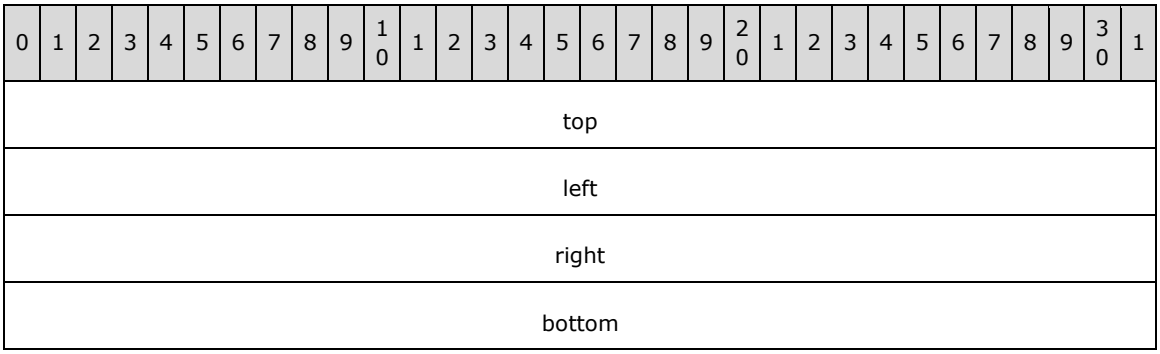
numer (4 bytes): A signed integer that specifies the numerator portion of this ratio.

denom (4 bytes): A signed integer that specifies the denominator portion of this ratio. It MUST NOT be 0x00000000.

2.12.7 RectStruct

Referenced by: OfficeArtClientAnchorData

A structure that specifies a rectangle.



top (4 bytes): A signed integer that specifies the minimum *y*-value of the rectangle.

left (4 bytes): A signed integer that specifies the minimum *x*-value of the rectangle.

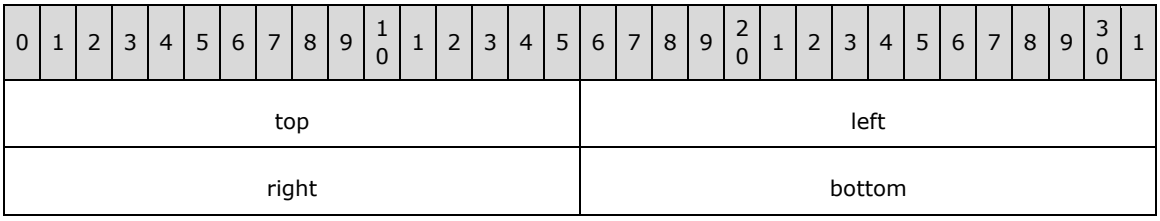
right (4 bytes): A signed integer that specifies the maximum *x*-value of the rectangle.

bottom (4 bytes): A signed integer that specifies the maximum *y*-value of the rectangle.

2.12.8 SmallRectStruct

Referenced by: OfficeArtClientAnchorData

A structure that specifies a small rectangle.



top (2 bytes): A signed integer that specifies the minimum *y*-value of the rectangle.

left (2 bytes): A signed integer that specifies the minimum *x*-value of the rectangle.

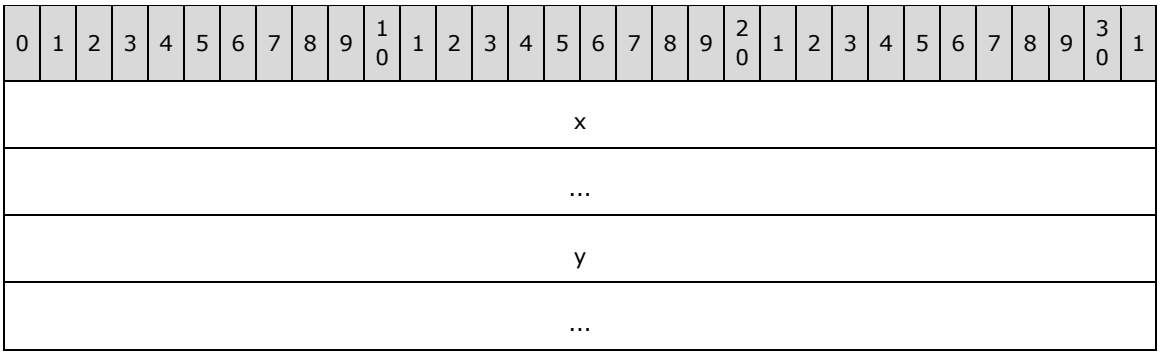
right (2 bytes): A signed integer that specifies the maximum *x*-value of the rectangle.

bottom (2 bytes): A signed integer that specifies the maximum *y*-value of the rectangle.

2.12.9 ScalingStruct

Referenced by: NoZoomViewInfoAtom, ZoomViewInfoAtom

A structure that specifies two-dimensional scaling.



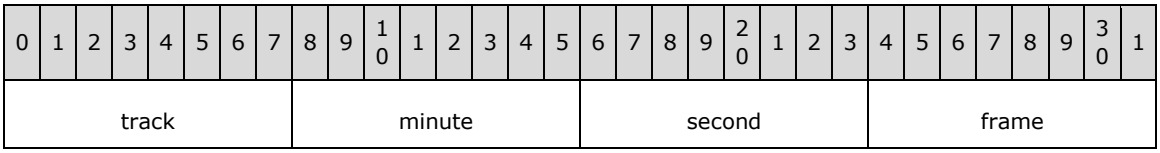
x (8 bytes): A **RatioStruct** structure (section 2.12.6) that specifies the scale to apply along the *x*-axis.

y (8 bytes): A **RatioStruct** structure that specifies the scale to apply along the *y*-axis.

2.12.10 TmsfTimeStruct

Referenced by: ExCDAudioAtom

A structure that specifies CD (compact disc) audio time in terms of tracks, minutes, seconds, and frames.



track (1 byte): An unsigned integer that specifies the track number. It MUST be greater than 0x00 and less than or equal to 0x64.

minute (1 byte): An unsigned integer that specifies the number of minutes. It MUST be less than or equal to 0x3C.

second (1 byte): An unsigned integer that specifies the number of seconds. It MUST be less than 0x3C.

frame (1 byte): An unsigned integer that specifies the frame number. It MUST be less than 0x4A.

2.13 Enumerations

2.13.1 AnimAfterEffectEnum

Referenced by: AnimationInfoAtom

An enumeration that specifies behavior types of shapes or text after animation effects are complete.

Name	Value	Meaning
AI_NoAfterEffect	0x00	No further change to the animated object after the animation is complete.
AI_Dim	0x01	Change the animated object to a specified color after the animation is complete.
AI_Hide	0x02	Hide the animated object on the next mouse click.

Name	Value	Meaning
AI_HideImmediately	0x03	Hide the animated object immediately after the animation is complete.

2.13.2 AnimBuildTypeEnum

Referenced by: AnimationInfoAtom

An enumeration that specifies animation build types.

Name	Value	Meaning
BT_FollowMaster	0xFE	The shape follows the build type of the placeholder shape on its main master slide or its title master slide.
BT_NoBuild	0x00	No build for the shape.
BT_OneBuild	0x01	The shape animates in its entirety.
BT_Level1Build	0x02	Each paragraph of level 1 animates separately, and paragraphs of all other levels animate at the same time as their level 1 paragraphs.
BT_Level2Build	0x03	Each paragraph from level 1 to level 2 animates separately, and paragraphs of all other levels animate at the same time as their level 2 paragraphs.
BT_Level3Build	0x04	Each paragraph from level 1 to level 3 animates separately, and paragraphs of all other levels animate at the same time as their level 3 paragraphs.
BT_Level4Build	0x05	Each paragraph from level 1 to level 4 animates separately, and paragraphs of level 5 animate at the same time as their level 4 paragraphs.
BT_Level5Build	0x06	Each paragraph from level 1 to level 5 animates separately.
BT_GraphBySeries	0x07	Each series animates separately, and all elements in each series animate at the same time.
BT_GraphByCategory	0x08	Each category animates separately, and all elements in each category animate at the same time.
BT_GraphByElementInSeries	0x09	Elements in the chart animate in the following order: Element in series 1 and category 1 Element in series 1 and category 2 Element in series 1 and category 3 … Element in series 2 and category 1 Element in series 2 and category 2 Element in series 2 and category 3 … Element in series 3 and category 1 Element in series 3 and category 2 Element in series 3 and category 3 …
BT_GraphByElementInCategory	0x0A	Elements in the chart animate in the following order:

Name	Value	Meaning
		Element in category 1 and series 1 Element in category 1 and series 2 Element in category 1 and series 3 … Element in category 2 and series 1 Element in category 2 and series 2 Element in category 2 and series 3 … Element in category 3 and series 1 Element in category 3 and series 2 Element in category 3 and series 3 …

2.13.3 BuildTypeEnum

Referenced by: BuildAtom

An enumeration that specifies different types of builds.

Name	Value	Meaning
TL_BuildParagraph	0x00000001	Paragraph build type.
TL_BuildChart	0x00000002	Chart build type.
TL_BuildDiagram	0x00000003	Diagram build type.

2.13.4 ChartBuildEnum

Referenced by: ChartBuildAtom

An enumeration that specifies different types of chart builds.

Name	Value	Meaning
TLCB_AsOneObject	0x00000000	Chart animates in its entirety.
TLCB_BySeries	0x00000001	Each series animates separately, and all elements in each series animate at the same time.
TLCB_ByCategory	0x00000002	Each category animates separately, and all elements in each category animate at the same time.
TLCB_ByElementInSeries	0x00000003	Elements in the chart animate in the following order: Element in series 1 and category 1 Element in series 1 and category 2 Element in series 1 and category 3 … Element in series 2 and category 1 Element in series 2 and category 2 Element in series 2 and category 3 … Element in series 3 and category 1

Name	Value	Meaning
		Element in series 3 and category 2 Element in series 3 and category 3 …
TLCB_ByElementInCategory	0x00000004	Elements in the chart animate in the following order: Element in category 1 and series 1 Element in category 1 and series 2 Element in category 1 and series 3 … Element in category 2 and series 1 Element in category 2 and series 2 Element in category 2 and series 3 … Element in category 3 and series 1 Element in category 3 and series 2 Element in category 3 and series 3 …
TLCB_Custom	0x00000005	Custom chart build type.

2.13.5 ColorModeEnum

Referenced by: PrintOptionsAtom

An enumeration that specifies how colors are printed.

Name	Value	Meaning
CM_BlackAndWhite	0x00	Every color should be represented as black only or white only.
CM_Grayscale	0x01	Every color should be represented by its corresponding shade of gray.
CM_Color	0x02	No processing is done on the colors before sending them to the printer.

2.13.6 ConditionEnum

Referenced by: TimeConditionContainer

An enumeration that specifies the type of a time condition.

Name	Value	Meaning
TL_CT_None	0x00000001	None.
TL_CT_Begin	0x00000002	Begin condition that specifies when a time node will be activated.
TL_CT_End	0x00000003	End condition that specifies when a time node will be deactivated.
TL_CT_Next	0x00000004	Next condition that specifies when the next child time node of a sequential time node will be activated.
TL_CT_Previous	0x00000005	Previous condition that specifies when the previous child time node of a sequential time node will be activated.
TL_CT_EndSync	0x00000006	EndSync condition that specifies how to synchronize the stopping of the child

Name	Value	Meaning
		nodes of a time node.

2.13.7 DiagramBuildEnum

Referenced by: DiagramBuildAtom

An enumeration that specifies the animation diagram build type to be applied to a diagram.

Name	Value	Meaning
TLDB_AsOneObject	0x00000000	The diagram and all corresponding parts animate as one graphical object.
TLDB_DepthByNode	0x00000001	The root shape of the diagram animates first, followed by its branches from left to right. For each branch, the root shape of the branch animates first, followed by the branches of this branch. This process is recursive for each shape in the diagram.
TLDB_DepthByBranch	0x00000002	The root shape of the diagram animates first, followed by its branches from left to right; and each of its branches animates as one graphical object.
TLDB_BreadthByNode	0x00000003	The root shape of the diagram animates first, followed by its levels from top to bottom. Shapes in each level animate separately from left to right.
TLDB_BreadthByLevel	0x00000004	The root shape of the diagram animates first, followed by its levels from top to bottom. Each level animates as one graphical object.
TLDB_ClockWise	0x00000005	Shapes in the diagram animate in the clockwise direction.
TLDB_ClockWiseIn	0x00000006	Shapes in the diagram animate in the clockwise direction. Shapes animate inwardly starting from the outermost ring.
TLDB_ClockWiseOut	0x00000007	Shapes in the diagram animate in the clockwise direction. Shapes animate outwardly starting from the innermost ring.
TLDB_CounterClockWise	0x00000008	Shapes in the diagram animate in the counterclockwise direction.
TLDB_CounterClockWiseIn	0x00000009	Shapes in the diagram animate in the counterclockwise direction. Shapes animate inwardly starting from the outermost ring.
TLDB_CounterClockWiseOut	0x0000000A	Shapes in the diagram animate in the counterclockwise direction. Shapes animate outwardly starting from the innermost ring.
TLDB_InByRing	0x0000000B	Rings in the diagram animate from the outside to the inside. All shapes in each ring animate at the same time.
TLDB_OutByRing	0x0000000C	Rings in the diagram animate from the inside to the outside. All shapes in each ring animate at the same time.
TLDB_Up	0x0000000D	Shapes in the diagram animate from bottom to top.
TLDB_Down	0x0000000E	Shapes in the diagram animate from top to bottom.
TLDB_AllAtOnce	0x0000000F	All shapes in the diagram animate at the same time.

Name	Value	Meaning
TLDB_Custom	0x00000010	Shapes in the diagram animate in a custom way not otherwise specified by one of the allowed diagram build types.

2.13.8 DiffTypeEnum

Referenced by: DiffRecordHeaders

An enumeration that specifies different types of document changes made by a reviewer.

Name	Value	Meaning
Diff_DocDiff	0x00000000	Document level change.
Diff_SlideDiff	0x00000002	Slide change.
Diff_MainMasterDiff	0x00000003	Main master change.
Diff_SlideListDiff	0x00000004	Slide list change.
Diff_MasterListDiff	0x00000005	Master list change.
Diff_ShapeListDiff	0x00000006	Shape list change.
Diff_ShapeDiff	0x00000007	Shape change.
Diff_TextDiff	0x00000009	Text change.
Diff_NotesDiff	0x0000000A	Notes change.
Diff_SlideShowDiff	0x0000000B	Slide show change.
Diff_HeaderFooterDiff	0x0000000C	Header footer change.
Diff_NamedShowDiff	0x0000000E	Named show change.
Diff_NamedShowListDiff	0x0000000F	Named show list change.
Diff_RecolorInfoDiff	0x00000012	Recolor info change.
Diff_ExternalObjectDiff	0x00000013	External object change.
Diff_TableListDiff	0x00000015	Table list change.
Diff_TableDiff	0x00000016	Table change.
Diff_InteractiveInfoDiff	0x00000017	Interactive information change.

2.13.9 ElementTypeEnum

Referenced by: VisualShapeChartElementAtom, VisualShapeGeneralAtom, VisualSoundAtom

An enumeration that specifies the element type of an animation target.

Name	Value	Meaning
TL_ET_ShapeType	0x00000001	The animation targets a shape or some part of a shape.
TL_ET_SoundType	0x00000002	The animation targets a sound file that does not correspond to a shape.

2.13.10 ExColorFollowEnum

Referenced by: ExOleEmbedAtom

An enumeration that specifies how an OLE object follows the color scheme.

Name	Value	Meaning
ExColor_FollowNone	0x00000000	Does not follow the color scheme.
ExColor_FollowScheme	0x00000001	Follows the color scheme.
ExColor_FollowTextAndBackground	0x00000002	Follows only the text and background colors of the color scheme.

2.13.11 ExOleObjSubTypeEnum

Referenced by: ExOleObjAtom

An enumeration that specifies the subtype of an OLE object based on its **ProgID** (described in [MSDN-COM]).

Name	Value	Meaning
ExOleSub_Default	0x00000000	Used when none of the following apply.
ExOleSub_Clipart	0x00000001	MS_ClipArt_Gallery
ExOleSub_WordDoc	0x00000002	Word.Document or Word.DocumentMacroEnabled
ExOleSub_Excel	0x00000003	Excel.Sheet, Excel.SheetMacroEnabled or Excel.SheetBinaryMacroEnabled
ExOleSub_Graph	0x00000004	MSGraph.Chart or MSGraph
ExOleSub_OrgChart	0x00000005	OrgChart, MSOrgChart or OrgPlusWOPX
ExOleSub_Equation	0x00000006	Equations or Equation
ExOleSub_WordArt	0x00000007	MSWordArt
ExOleSub_Sound	0x00000008	SoundRec
ExOleSub_Project	0x0000000C	MSProject
ExOleSub_NoteIt	0x0000000D	Note-It
ExOleSub_ExcelChart	0x0000000E	Excel.Chart
ExOleSub_MediaPlayer	0x0000000F	MPlayer, MIDFile or AVIFile
ExOleSub_WordPad	0x00000010	WordPad.Document
ExOleSub_Visio	0x00000011	Visio.Drawing<113>
ExOleSub_WordODF	0x00000012	Word.OpenDocumentText<114>
ExOleSub_ExcelODF	0x00000013	Excel.OpenDocumentSpreadsheet<115>
ExOleSub_PPTODF	0x00000014	PowerPoint.OpenDocumentPresentation<116>

2.13.12 ExOleObjTypeEnum

Referenced by: ExOleObjAtom

An enumeration that specifies the type of an OLE object.

Name	Value	Meaning
ExOle_Embedded	0x00000000	An embedded OLE object; the object is serialized and saved within the file.
ExOle_Link	0x00000001	A linked OLE object; the object is saved outside of the file.
ExOle_Control	0x00000002	The OLE object is an ActiveX control.

2.13.13 InteractiveInfoActionEnum

Referenced by: InteractiveInfoAtom

An enumeration that specifies an action that can be performed when interacting with an object during a slide show.

Name	Value	Meaning
II_NoAction	0x00	No effect.
II_MacroAction	0x01	A macro is executed.
II_RunProgramAction	0x02	A program is run.
II_JumpAction	0x03	The current presentation slide of the slide show jumps to another presentation slide in the same presentation.
II_HyperlinkAction	0x04	A URL is executed.
II_OLEAction	0x05	An OLE action (only valid if the object this applies to is an OLE embedded object).
II_MediaAction	0x06	A media object is played.
II_CustomShowAction	0x07	A named show is displayed.

2.13.14 InteractiveInfoJumpEnum

Referenced by: InteractiveInfoAtom

An enumeration that specifies a location relative to the currently-displayed presentation slide in the slide show.

Name	Value	Meaning
II_NoJump	0x00	No change.
II_NextSlide	0x01	The next slide.
II_PreviousSlide	0x02	The previous slide.
II_FirstSlide	0x03	The first slide.
II_LastSlide	0x04	The last slide.

Name	Value	Meaning
II_LastSlideViewed	0x05	The last viewed slide.
II_EndShow	0x06	The end of show slide (a virtual slide displayed after the last slide).

2.13.15 LinkToEnum

Referenced by: InteractiveInfoAtom

An enumeration that specifies how the action of a hyperlink is interpreted. All locations are relative to the currently-displayed presentation slide in the slide show.

Name	Value	Meaning
LT_NextSlide	0x00	The next slide.
LT_PreviousSlide	0x01	The previous slide.
LT_FirstSlide	0x02	The first slide.
LT_LastSlide	0x03	The last slide.
LT_CustomShow	0x06	A named show.
LT_SlideNumber	0x07	A specific slide number.
LT_Url	0x08	A **Uniform Resource Locator (URL)**.
LT_OtherPresentation	0x09	Another presentation file.
LT_OtherFile	0x0A	Another file that is not necessarily a presentation.
LT_Nil	0xFF	The hyperlink is not valid.

2.13.16 NormalViewSetBarStates

Referenced by: NormalViewSetInfoAtom

An enumeration that specifies different states of a region of a view.

Name	Value	Meaning
BS_Minimized	0x00	The region occupies a minimal area of the view.
BS_Restored	0x01	The region has an intermediate size.
BS_Maximized	0x02	The region occupies a maximal area of the view.

2.13.17 OLEVerbEnum

Referenced by: AnimationInfoAtom, InteractiveInfoAtom

An enumeration that specifies the identifier of an **OLE verb**. Because this enumeration refers to values defined by the OLE object that it is linked to, the sample values listed in the table are placeholders that specify which command to run. The actual number of verbs depends on the OLE object itself.

Name	Value	Meaning
OV_Primary	0x00	The primary verb is to be used.
OV_Secondary	0x01	The secondary verb is to be used.
OV_Tertiary	0x02	The tertiary verb is to be used.

2.13.18 ParaBuildEnum

Referenced by: ParaBuildAtom

An enumeration that specifies the animation paragraph build type that is to be applied to the paragraphs of the shape.

Name	Value	Meaning
TLPB_AllAtOnce	0x00000000	All paragraphs in the shape animate at the same time.
TLPB_BuildByNthLevel	0x00000001	Paragraph levels 1 to *n* – 1 in the shape animate separately. All paragraph levels *n* or greater animate at the same time.
TLPB_CustomBuild	0x00000002	Applies a custom animation paragraph build type to the paragraphs of the shape.
TLPB_AsAWhole	0x00000003	The shape and all paragraphs within the shape animate as one graphical object.

2.13.19 PhotoAlbumFrameShapeEnum

Referenced by: PhotoAlbumInfo10Atom

An enumeration that specifies how the frames around the pictures in the photo album are drawn. Some frames are created by cropping the photos to a certain shape, and others involve putting a frame image on top of them without modifying them.

Name	Value	Meaning
PA_Rectangle	0x0000	The pictures are drawn normally.
PA_RoundedRectangle	0x0001	The pictures are drawn with their edges cropped such that the shape of the frame of the pictures is a rounded rectangle.
PA_Beveled	0x0002	The pictures are drawn to look like the frame has a beveled edge.
PA_Oval	0x0003	The pictures are drawn with their edges cropped such that the shape of the frame of the pictures is an oval.
PA_Octagon	0x0004	The pictures are drawn with triangular shapes covering the four corners of the frame.
PA_Cross	0x0005	The pictures are drawn with square shapes covering the four corners of the frame.
PA_Plaque	0x0006	The pictures are drawn with rounded shapes covering the four corners of the frame.

2.13.20 PhotoAlbumLayoutEnum

Referenced by: PhotoAlbumInfo10Atom

An enumeration that specifies how the pictures are arranged on each presentation slide.

Name	Value	Meaning
PA_FitToSlide	0x00	Each presentation slide contains one picture that is scaled as large as will fit within the bounds of the slides while still preserving the aspect ratio.
PA_OnePicture	0x01	Each presentation slide contains one picture.
PA_TwoPictures	0x02	Each presentation slide contains two pictures.
PA_FourPictures	0x03	Each presentation slide contains four pictures
PA_OnePictureAndTitle	0x04	Each presentation slide contains one picture and a title placeholder shape.
PA_TwoPicturesAndTitle	0x05	Each presentation slide contains two pictures and a title placeholder shape.
PA_FourPicturesAndTitle	0x06	Each presentation slide contains four pictures and a title placeholder shape.

2.13.21 PlaceholderEnum

Referenced by: PlaceholderAtom, RoundTripHFPlaceholder12Atom, RoundTripNewPlaceholderId12Atom, SlideAtom

An enumeration that specifies the type of a placeholder shape. The meaning of each enumeration value is further specified in the PlaceholderAtom record. This enumeration is also used to define a slide layout as described in the SlideAtom record.

Name	Value	Meaning
PT_None	0x00	No placeholder shape.
PT_MasterTitle	0x01	Master title text placeholder shape.
PT_MasterBody	0x02	Master body text placeholder shape.
PT_MasterCenterTitle	0x03	Master center title text placeholder shape.
PT_MasterSubTitle	0x04	Master sub-title text placeholder shape.
PT_MasterNotesSlideImage	0x05	Master notes slide image placeholder shape.
PT_MasterNotesBody	0x06	Master notes body text placeholder shape.
PT_MasterDate	0x07	Master date placeholder shape.
PT_MasterSlideNumber	0x08	Master slide number placeholder shape.
PT_MasterFooter	0x09	Master footer placeholder shape.
PT_MasterHeader	0x0A	Master header placeholder shape.
PT_NotesSlideImage	0x0B	Notes slide image placeholder shape.
PT_NotesBody	0x0C	Notes body text placeholder shape.

Name	Value	Meaning
PT_Title	0x0D	Title text placeholder shape.
PT_Body	0x0E	Body text placeholder shape.
PT_CenterTitle	0x0F	Center title text placeholder shape.
PT_SubTitle	0x10	Sub-title text placeholder shape.
PT_VerticalTitle	0x11	Vertical title text placeholder shape.
PT_VerticalBody	0x12	Vertical body text placeholder shape.
PT_Object	0x13	Object placeholder shape.
PT_Graph	0x14	Graph object placeholder shape.
PT_Table	0x15	Table object placeholder shape.
PT_ClipArt	0x16	Clipart object placeholder shape.
PT_OrgChart	0x17	Organization chart object placeholder shape.
PT_Media	0x18	Media object placeholder shape.
PT_VerticalObject	0x19	Vertical object placeholder shape.
PT_Picture	0x1A	Picture object placeholder shape.

2.13.22 PlaceholderSize

Referenced by: PlaceholderAtom

An enumeration that specifies the preferred size of a placeholder shape. The size is relative to the size of the master body text placeholder shape.

Name	Value	Meaning
PS_Full	0x00	The full size of the master body text placeholder shape.
PS_Half	0x01	Half of the size of the master body text placeholder shape.
PS_Quarter	0x02	A quarter of the size of the master body text placeholder shape.

2.13.23 PrintWhatEnum

Referenced by: PrintOptionsAtom

An enumeration that specifies which aspect of the presentation to print.

Name	Value	Meaning
PW_Slides	0x00	The presentation slides.
PW_BuildSlides	0x01	The presentation slides plus extra images showing the steps of animations.
PW_Handouts2	0x02	A layout optimized for handout slides where two slides are shown per page.
PW_Handouts3	0x03	A layout optimized for handout slides where three slides are shown per page.

Name	Value	Meaning
PW_Handouts6	0x04	A layout optimized for handout slides where six slides are shown per page.
PW_Notes	0x05	The presentation slides plus the attached notes.
PW_Outline	0x06	The text outline of the presentation.
PW_Handouts4	0x07	A layout optimized for handout slides where four slides are shown per page.
PW_Handouts9	0x08	A layout optimized for handout slides where nine slides are shown per page.
PW_Handouts1	0x09	A layout optimized for handout slides where one slide is shown per page.

2.13.24 RecordType

Referenced by: RecordHeader

An enumeration that specifies the record type of an atom record or a container record.

Name	Value	Meaning
RT_Document	0x03E8	Specifies a **DocumentContainer** (section 2.4.1).
RT_DocumentAtom	0x03E9	Specifies a **DocumentAtom** (section 2.4.2).
RT_EndDocumentAtom	0x03EA	Specifies a **EndDocumentAtom** record (section 2.4.13).
RT_Slide	0x03EE	Specifies a **SlideContainer** (section 2.5.1).
RT_SlideAtom	0x03EF	Specifies a SlideAtom.
RT_Notes	0x03F0	Specifies a **NotesContainer** (section 2.5.6).
RT_NotesAtom	0x03F1	Specifies a NotesAtom.
RT_Environment	0x03F2	Specifies a **DocumentTextInfoContainer** record (section 2.9.1).
RT_SlidePersistAtom	0x03F3	Specifies a **MasterPersistAtom** (section 2.4.14.2), **SlidePersistAtom** (section 2.4.14.5), or **NotesPersistAtom** (section 2.4.14.7).
RT_MainMaster	0x03F8	Specifies a **MainMasterContainer** (section 2.5.3).
RT_SlideShowSlideInfoAtom	0x03F9	Specifies a SlideShowSlideInfoAtom.
RT_SlideViewInfo	0x03FA	Specifies a **SlideViewInfoContainer** (section 2.4.21.9) or **NotesViewInfoContainer** (section 2.4.21.12).
RT_GuideAtom	0x03FB	Specifies a GuideAtom.
RT_ViewInfoAtom	0x03FD	Specifies a ZoomViewInfoAtom or NoZoomViewInfoAtom.
RT_SlideViewInfoAtom	0x03FE	Specifies a SlideViewInfoAtom.
RT_VbaInfo	0x03FF	Specifies a **VBAInfoContainer** (section 2.4.10).
RT_VbaInfoAtom	0x0400	Specifies a VBAInfoAtom.

Name	Value	Meaning
RT_SlideShowDocInfoAtom	0x0401	Specifies a **SlideShowDocInfoAtom** record (section 2.6.1).
RT_Summary	0x0402	Specifies a **SummaryContainer** record (section 2.4.22.3).
RT_DocRoutingSlipAtom	0x0406	Specifies a **DocRoutingSlipAtom** record (section 2.11.1).
RT_OutlineViewInfo	0x0407	Specifies an **OutlineViewInfoContainer** (section 2.4.21.6).
RT_SorterViewInfo	0x0408	Specifies a **SorterViewInfoContainer** record (section 2.4.21.13).
RT_ExternalObjectList	0x0409	Specifies an **ExObjListContainer** (section 2.10.1).
RT_ExternalObjectListAtom	0x040A	Specifies an ExObjListAtom.
RT_DrawingGroup	0x040B	Specifies a **DrawingGroupContainer** (section 2.4.3).
RT_Drawing	0x040C	Specifies a **DrawingContainer** (section 2.5.13).
RT_GridSpacing10Atom	0x040D	Specifies a GridSpacing2.4.21.1Atom.
RT_RoundTripTheme12Atom	0x040E	Specifies a RoundTripThemeAtom.
RT_RoundTripColorMapping12Atom	0x040F	Specifies a RoundTripColorMappingAtom.
RT_NamedShows	0x0410	Specifies a **NamedShowsContainer** (section 2.6.2).
RT_NamedShow	0x0411	Specifies a NamedShowContainer.
RT_NamedShowSlidesAtom	0x0412	Specifies a NamedShowSlidesAtom.
RT_NotesTextViewInfo9	0x0413	Specifies a **NotesTextViewInfoContainer** (section 2.4.21.4).
RT_NormalViewSetInfo9	0x0414	Specifies a **NormalViewSetInfoContainer** (section 2.4.21.2).
RT_NormalViewSetInfo9Atom	0x0415	Specifies a NormalViewSetInfoAtom.
RT_RoundTripOriginalMainMasterId12Atom	0x041C	Specifies a RoundTripOriginalMainMasterId2.11.20Atom.
RT_RoundTripCompositeMasterId12Atom	0x041D	Specifies a RoundTripCompositeMasterId2.11.10Atom.
RT_RoundTripContentMasterInfo12Atom	0x041E	Specifies a RoundTripContentMasterInfo2.11.12Atom.
RT_RoundTripShapeId12Atom	0x041F	Specifies a RoundTripShapeId2.11.22Atom.
RT_RoundTripHFPlaceholder12Atom	0x0420	Specifies a RoundTripHFPlaceholder2.11.16Atom.
RT_RoundTripContentMasterId12Atom	0x0422	Specifies a RoundTripContentMasterId2.11.11Atom.
RT_RoundTripOArtTextStyles12Atom	0x0423	Specifies a RoundTripOArtTextStyles2.11.19Atom.
RT_RoundTripHeaderFooterDefaults12Ato	0x0424	Specifies a

Name	Value	Meaning
m		RoundTripHeaderFooterDefaults2.11.15Atom.
RT_RoundTripDocFlags12Atom	0x0425	Specifies a RoundTripDocFlags2.11.14Atom.
RT_RoundTripShapeCheckSumForCL12Atom	0x0426	Specifies a RoundTripShapeCheckSumForCustomLayouts2.11.21.
RT_RoundTripNotesMasterTextStyles12Atom	0x0427	Specifies a RoundTripNotesMasterTextStyles2.11.18Atom.
RT_RoundTripCustomTableStyles12Atom	0x0428	Specifies a **RoundTripCustomTableStyles12Atom** record (section 2.11.13).
RT_List	0x07D0	Specifies a **DocInfoListContainer** (section 2.4.4).
RT_FontCollection	0x07D5	Specifies a **FontCollectionContainer** (section 2.9.8).
RT_FontCollection10	0x07D6	Specifies a **FontCollection10Container** (section 2.9.11).
RT_BookmarkCollection	0x07E3	Specifies a BookmarkCollectionContainer.
RT_SoundCollection	0x07E4	Specifies a **SoundCollectionContainer** record (section 2.4.16.1).
RT_SoundCollectionAtom	0x07E5	Specifies a SoundCollectionAtom.
RT_Sound	0x07E6	Specifies a **SoundContainer** (section 2.4.16.3).
RT_SoundDataBlob	0x07E7	Specifies a SoundDataBlob.
RT_BookmarkSeedAtom	0x07E9	Specifies a BookmarkSeedAtom.
RT_ColorSchemeAtom	0x07F0	Specifies a SlideSchemeColorSchemeAtom or SchemeListElementColorSchemeAtom.
RT_BlipCollection9	0x07F8	Specifies a **BlipCollection9Container** (section 2.9.72).
RT_BlipEntity9Atom	0x07F9	Specifies a BlipEntityAtom.
RT_ExternalObjectRefAtom	0x0BC1	Specifies an ExObjRefAtom.
RT_PlaceholderAtom	0x0BC3	Specifies a PlaceholderAtom.
RT_ShapeAtom	0x0BDB	Specifies a ShapeFlagsAtom.
RT_ShapeFlags10Atom	0x0BDC	Specifies a ShapeFlags2.7.6Atom.
RT_RoundTripNewPlaceholderId12Atom	0x0BDD	Specifies a RoundTripNewPlaceholderId2.11.17Atom.
RT_OutlineTextRefAtom	0x0F9E	Specifies an OutlineTextRefAtom.
RT_TextHeaderAtom	0x0F9F	Specifies a TextHeaderAtom.
RT_TextCharsAtom	0x0FA0	Specifies a TextCharsAtom.

Name	Value	Meaning
RT_StyleTextPropAtom	0x0FA1	Specifies a StyleTextPropAtom.
RT_MasterTextPropAtom	0x0FA2	Specifies a MasterTextPropAtom.
RT_TextMasterStyleAtom	0x0FA3	Specifies a TextMasterStyleAtom.
RT_TextCharFormatExceptionAtom	0x0FA4	Specifies a TextCFExceptionAtom.
RT_TextParagraphFormatExceptionAtom	0x0FA5	Specifies a TextPFExceptionAtom.
RT_TextRulerAtom	0x0FA6	Specifies a TextRulerAtom.
RT_TextBookmarkAtom	0x0FA7	Specifies a TextBookmarkAtom.
RT_TextBytesAtom	0x0FA8	Specifies a TextBytesAtom.
RT_TextSpecialInfoDefaultAtom	0x0FA9	Specifies a TextSIExceptionAtom.
RT_TextSpecialInfoAtom	0x0FAA	Specifies a TextSpecialInfoAtom.
RT_DefaultRulerAtom	0x0FAB	Specifies a DefaultRulerAtom.
RT_StyleTextProp9Atom	0x0FAC	Specifies a StyleTextProp2.9.67Atom.
RT_TextMasterStyle9Atom	0x0FAD	Specifies a TextMasterStyle2.9.37Atom.
RT_OutlineTextProps9	0x0FAE	Specifies an OutlineTextProps2.9.60Container.
RT_OutlineTextPropsHeader9Atom	0x0FAF	Specifies an OutlineTextPropsHeaderExAtom.
RT_TextDefaults9Atom	0x0FB0	Specifies a TextDefaults2.9.74Atom.
RT_StyleTextProp10Atom	0x0FB1	Specifies a StyleTextProp2.9.69Atom.
RT_TextMasterStyle10Atom	0x0FB2	Specifies a TextMasterStyle2.9.39Atom.
RT_OutlineTextProps10	0x0FB3	Specifies an OutlineTextProps2.9.63Container.
RT_TextDefaults10Atom	0x0FB4	Specifies a TextDefaults2.9.75Atom.
RT_OutlineTextProps11	0x0FB5	Specifies an OutlineTextProps2.9.65Container.
RT_StyleTextProp11Atom	0x0FB6	Specifies a StyleTextProp2.9.70Atom.
RT_FontEntityAtom	0x0FB7	Specifies a FontEntityAtom.
RT_FontEmbedDataBlob	0x0FB8	Specifies a FontEmbedDataBlob.
RT_CString	0x0FBA	Specifies a KinsokuLeadingAtom, KinsokuFollowingAtom, NamedShowNameAtom, MacroNameAtom, UncOrLocalPathAtom, MenuNameAtom, ProgIDAtom, ClipboardNameAtom, FriendlyNameAtom, TargetAtom, LocationAtom, ScreenTipAtom, PP2.7.18ShapeBinaryTagExtension, PP2.7.19ShapeBinaryTagExtension, PP2.7.20ShapeBinaryTagExtension, SlideNameAtom, TemplateNameAtom, PP2.5.23SlideBinaryTagExtension, PP2.5.24SlideBinaryTagExtension, Comment2.5.26AuthorAtom, Comment2.5.27TextAtom, Comment2.5.28AuthorInitialAtom,

Name	Value	Meaning
		PP2.5.34SlideBinaryTagExtension, SoundNameAtom, PP2.4.23.5DocBinaryTagExtension, FileNameAtom, NamedShowAtom, BCTitleAtom, BCDescriptionAtom, BCSpeakerAtom, BCContactAtom, BCRexServerNameAtom, BCEmailAddressAtom, BCEmailNameAtom, BCChatUrlAtom, BCArchiveDirAtom, BCNetShowFilesBaseDirAtom, BCNetShowFilesDirAtom, BCNetShowServerNameAtom, BCPptFilesBaseDirAtom, BCPptFilesDirAtom, BCPptFilesBaseUrlAtom, BCBroadcastDateTimeAtom, BCPresentationNameAtom, BCAsdFileNameAtom, BCEntryIDAtom, PP2.4.23.6DocBinaryTagExtension, AuthorNameAtom, CopyrightAtom, KeywordsAtom, ModifyPasswordAtom, ReviewerNameAtom, PP2.4.23.7DocBinaryTagExtension, PP2.4.23.8DocBinaryTagExtension, UserDateAtom, HeaderAtom, FooterAtom, BookmarkValueAtom, TagNameAtom, TagValueAtom, SoundExtensionAtom, SoundIdAtom, SoundBuiltinIdAtom, BCUserNameAtom, ServerIdAtom, or SlideLibUrlAtom.
RT_MetaFile	0x0FC1	Specifies a MetafileBlob.
RT_ExternalOleObjectAtom	0x0FC3	Specifies an **ExOleObjAtom** section 2.10.12).
RT_Kinsoku	0x0FC8	Specifies a **KinsokuContainer** (section 2.9.2) or **Kinsoku9Container** (section 2.9.6).
RT_Handout	0x0FC9	Specifies a **HandoutContainer** (section 2.5.8).
RT_ExternalOleEmbed	0x0FCC	Specifies an **ExOleEmbedContainer** (section 2.10.27).
RT_ExternalOleEmbedAtom	0x0FCD	Specifies an ExOleEmbedAtom.
RT_ExternalOleLink	0x0FCE	Specifies an **ExOleLinkContainer** (section 2.10.29).
RT_BookmarkEntityAtom	0x0FD0	Specifies a BookmarkEntityAtom or BookmarkEntityAtomContainer.
RT_ExternalOleLinkAtom	0x0FD1	Specifies a ExOleLinkAtom.
RT_KinsokuAtom	0x0FD2	Specifies a KinsokuAtom or Kinsoku2.9.7Atom.
RT_ExternalHyperlinkAtom	0x0FD3	Specifies an **ExHyperlinkAtom** (section 2.10.17**)** or ExHyperlinkRefAtom.
RT_ExternalHyperlink	0x0FD7	Specifies an ExHyperlinkContainer.
RT_SlideNumberMetaCharAtom	0x0FD8	Specifies a SlideNumberMCAtom.
RT_HeadersFooters	0x0FD9	Specifies a **SlideHeadersFootersContainer** (section 2.4.15.1), **NotesHeadersFootersContainer** (section 2.4.15.6), or PerSlideHeadersFootersContainer.
RT_HeadersFootersAtom	0x0FDA	Specifies a HeadersFootersAtom.

Name	Value	Meaning
RT_TextInteractiveInfoAtom	0x0FDF	Specifies a MouseClickTextInteractiveInfoAtom or MouseOverTextInteractiveInfoAtom.
RT_ExternalHyperlink9	0x0FE4	Specifies an ExHyperlink2.10.21Container.
RT_RecolorInfoAtom	0x0FE7	Specifies a RecolorInfoAtom.
RT_ExternalOleControl	0x0FEE	Specifies an **ExControlContainer** (section 2.10.10).
RT_SlideListWithText	0x0FF0	Specifies a **MasterListWithTextContainer** (section 2.4.14.1), **SlideListWithTextContainer** (section 2.4.14.3), or **NotesListWithTextContainer** (section 2.4.14.6).
RT_AnimationInfoAtom	0x0FF1	Specifies an AnimationInfoAtom.
RT_InteractiveInfo	0x0FF2	Specifies a MouseClickInteractiveInfoContainer or MouseOverInteractiveInfoContainer.
RT_InteractiveInfoAtom	0x0FF3	Specifies an InteractiveInfoAtom.
RT_UserEditAtom	0x0FF5	Specifies a **UserEditAtom** (section 2.3.3).
RT_CurrentUserAtom	0x0FF6	Specifies a CurrentUserAtom.
RT_DateTimeMetaCharAtom	0x0FF7	Specifies a DateTimeMCAtom.
RT_GenericDateMetaCharAtom	0x0FF8	Specifies a GenericDateMCAtom.
RT_HeaderMetaCharAtom	0x0FF9	Specifies a HeaderMCAtom.
RT_FooterMetaCharAtom	0x0FFA	Specifies a FooterMCAtom.
RT_ExternalOleControlAtom	0x0FFB	Specifies an ExControlAtom.
RT_ExternalMediaAtom	0x1004	Specifies an **ExMediaAtom** (section 2.10.6).
RT_ExternalVideo	0x1005	Specifies an ExVideoContainer.
RT_ExternalAviMovie	0x1006	Specifies an ExAviMovieContainer.
RT_ExternalMciMovie	0x1007	Specifies an ExMCIMovieContainer.
RT_ExternalMidiAudio	0x100D	Specifies an ExMIDIAudioContainer.
RT_ExternalCdAudio	0x100E	Specifies an ExCDAudioContainer.
RT_ExternalWavAudioEmbedded	0x100F	Specifies an ExWAVAudioEmbeddedContainer.
RT_ExternalWavAudioLink	0x1010	Specifies an ExWAVAudioLinkContainer.
RT_ExternalOleObjectStg	0x1011	Specifies an ExOleObjStgCompressedAtom, ExOleObjStgUncompressedAtom, VbaProjectStgCompressedAtom, VbaProjectStgUncompressedAtom, ExControlStgUncompressedAtom, or ExControlStgCompressedAtom.
RT_ExternalCdAudioAtom	0x1012	Specifies an ExCDAudioAtom.
RT_ExternalWavAudioEmbeddedAtom	0x1013	Specifies an ExWAVAudioEmbeddedAtom.

Name	Value	Meaning
RT_AnimationInfo	0x1014	Specifies an AnimationInfoContainer.
RT_RtfDateTimeMetaCharAtom	0x1015	Specifies a RTFDateTimeMCAtom.
RT_ExternalHyperlinkFlagsAtom	0x1018	Specifies an ExHyperlinkFlagsAtom.
RT_ProgTags	0x1388	Specifies a SlideProgTagsContainer, **DocProgTagsContainer** (section 2.4.23.1), or ShapeProgTagsContainer.
RT_ProgStringTag	0x1389	Specifies a ProgStringTagContainer.
RT_ProgBinaryTag	0x138A	Specifies a SlideProgBinaryTagContainer, DocProgBinaryTagContainer, or ShapeProgBinaryTagContainer.
RT_BinaryTagDataBlob	0x138B	Specifies a PP2.7.18ShapeBinaryTagExtension, PP2.7.19ShapeBinaryTagExtension, PP2.7.20ShapeBinaryTagExtension, PP2.4.23.5DocBinaryTagExtension, PP2.4.23.6DocBinaryTagExtension, PP2.4.23.7DocBinaryTagExtension, PP2.4.23.8DocBinaryTagExtension, PP2.5.23SlideBinaryTagExtension, PP2.5.24SlideBinaryTagExtension, PP2.5.34SlideBinaryTagExtension, or BinaryTagDataBlob.
RT_PrintOptionsAtom	0x1770	Specifies a **PrintOptionsAtom** record (section 2.4.12).
RT_PersistDirectoryAtom	0x1772	Specifies a **PersistDirectoryAtom** (section 2.3.4).
RT_PresentationAdvisorFlags9Atom	0x177A	Specifies a PresAdvisorFlags2.4.6Atom.
RT_HtmlDocInfo9Atom	0x177B	Specifies an HTMLDocInfo2.4.18.1Atom.
RT_HtmlPublishInfoAtom	0x177C	Specifies an HTMLPublishInfoAtom.
RT_HtmlPublishInfo9	0x177D	Specifies an HTMLPublishInfo2.4.18.2Container.
RT_BroadcastDocInfo9	0x177E	Specifies a BroadcastDocInfo2.4.17.1Container.
RT_BroadcastDocInfo9Atom	0x177F	Specifies a BroadcastDocInfoAtom.
RT_EnvelopeFlags9Atom	0x1784	Specifies an EnvelopeFlags2.11.3Atom.
RT_EnvelopeData9Atom	0x1785	Specifies an EnvelopeData2.11.4Atom.
RT_VisualShapeAtom	0x2AFB	Specifies a VisualSoundAtom, VisualShapeChartElementAtom, or VisualShapeGeneralAtom.
RT_HashCodeAtom	0x2B00	Specifies a HashCode2.8.3Atom.
RT_VisualPageAtom	0x2B01	Specifies a VisualPageAtom.
RT_BuildList	0x2B02	Specifies a BuildListContainer.
RT_BuildAtom	0x2B03	Specifies a BuildAtom.
RT_ChartBuild	0x2B04	Specifies a ChartBuildContainer.

Name	Value	Meaning
RT_ChartBuildAtom	0x2B05	Specifies a ChartBuildAtom.
RT_DiagramBuild	0x2B06	Specifies a DiagramBuildContainer.
RT_DiagramBuildAtom	0x2B07	Specifies a DiagramBuildAtom.
RT_ParaBuild	0x2B08	Specifies a ParaBuildContainer.
RT_ParaBuildAtom	0x2B09	Specifies a ParaBuildAtom.
RT_LevelInfoAtom	0x2B0A	Specifies a LevelInfoAtom.
RT_RoundTripAnimationAtom12Atom	0x2B0B	Specifies a RoundTripAnimationAtom.
RT_RoundTripAnimationHashAtom12Atom	0x2B0D	Specifies a RoundTripAnimationHashAtom.
RT_Comment10	0x2EE0	Specifies a Comment2.5.25Container.
RT_Comment10Atom	0x2EE1	Specifies a Comment2.5.29Atom.
RT_CommentIndex10	0x2EE4	Specifies a CommentIndex2.4.19.1Container.
RT_CommentIndex10Atom	0x2EE5	Specifies a CommentIndex2.4.19.3Atom.
RT_LinkedShape10Atom	0x2EE6	Specifies a LinkedShape2.5.33Atom.
RT_LinkedSlide10Atom	0x2EE7	Specifies a LinkedSlide2.5.32Atom.
RT_SlideFlags10Atom	0x2EEA	Specifies a SlideFlags2.5.30Atom.
RT_SlideTime10Atom	0x2EEB	Specifies a SlideTime2.5.31Atom.
RT_DiffTree10	0x2EEC	Specifies a DiffTree2.4.20.5Container.
RT_Diff10	0x2EED	Specifies a DocDiff2.4.20.8Container, HeaderFooterDiffContainer, NamedShowListDiffContainer, NamedShowDiffContainer, SlideListDiffContainer, MasterListDiffContainer, MainMasterDiffContainer, SlideDiffContainer, ShapeListDiffContainer, ShapeDiffContainer, TextDiffContainer, RecolorInfoDiffContainer, ExternalObjectDiffContainer, InteractiveInfoDiffContainer, TableDiffContainer, SlideShowDiffContainer, NotesDiffContainer, or TableListDiffContainer.
RT_Diff10Atom	0x2EEE	Specifies a DocDiff2.4.20.8Container, HeaderFooterDiffContainer, NamedShowListDiffContainer, NamedShowDiffContainer, SlideListDiffContainer, MasterListDiffContainer, MainMasterDiffContainer, SlideDiffContainer, ShapeListDiffContainer, ShapeDiffContainer, TextDiffContainer, RecolorInfoDiffContainer, ExternalObjectDiffContainer, InteractiveInfoDiffContainer, TableDiffContainer, SlideShowDiffContainer, NotesDiffContainer, or TableListDiffContainer.
RT_SlideListTableSize10Atom	0x2EEF	Specifies a SlideListTableSize2.4.20.3Atom.
RT_SlideListEntry10Atom	0x2EF0	Specifies a SlideListEntry2.4.20.4Atom.

Name	Value	Meaning
RT_SlideListTable10	0x2EF1	Specifies a SlideListTable2.4.20.2Container.
RT_CryptSession10Container	0x2F14	Specifies a **CryptSession10Container** (section 2.3.7).
RT_FontEmbedFlags10Atom	0x32C8	Specifies a FontEmbedFlags2.9.12Atom.
RT_FilterPrivacyFlags10Atom	0x36B0	Specifies a FilterPrivacyFlags2.4.8Atom.
RT_DocToolbarStates10Atom	0x36B1	Specifies a DocToolbarStates2.4.20.1Atom.
RT_PhotoAlbumInfo10Atom	0x36B2	Specifies a PhotoAlbumInfo2.4.9Atom.
RT_SmartTagStore11Container	0x36B3	Specifies a **SmartTagStore11Container** (section 2.11.28).
RT_RoundTripSlideSyncInfo12	0x3714	Specifies a RoundTripSlideSyncInfo2.11.23Container.
RT_RoundTripSlideSyncInfoAtom12	0x3715	Specifies a SlideSyncInfoAtom2.11.26.
RT_TimeConditionContainer	0xF125	Specifies a **TimeConditionContainer** (section 2.8.75).
RT_TimeNode	0xF127	Specifies a TimeNodeAtom.
RT_TimeCondition	0xF128	Specifies a TimeConditionAtom.
RT_TimeModifier	0xF129	Specifies a TimeModifierAtom.
RT_TimeBehaviorContainer	0xF12A	Specifies a **TimeBehaviorContainer** (section 2.8.34).
RT_TimeAnimateBehaviorContainer	0xF12B	Specifies a **TimeAnimateBehaviorContainer** (section 2.8.29).
RT_TimeColorBehaviorContainer	0xF12C	Specifies a **TimeColorBehaviorContainer** (section 2.8.52).
RT_TimeEffectBehaviorContainer	0xF12D	Specifies a **TimeEffectBehaviorContainer** (section 2.8.61).
RT_TimeMotionBehaviorContainer	0xF12E	Specifies a **TimeMotionBehaviorContainer** (section 2.8.63).
RT_TimeRotationBehaviorContainer	0xF12F	Specifies a **TimeRotationBehaviorContainer** (section 2.8.65).
RT_TimeScaleBehaviorContainer	0xF130	Specifies a **TimeScaleBehaviorContainer** (section 2.8.67).
RT_TimeSetBehaviorContainer	0xF131	Specifies a **TimeSetBehaviorContainer** (section 2.8.69).
RT_TimeCommandBehaviorContainer	0xF132	Specifies a **TimeCommandBehaviorContainer** (section 2.8.71).
RT_TimeBehavior	0xF133	Specifies a TimeBehaviorAtom.
RT_TimeAnimateBehavior	0xF134	Specifies a TimeAnimateBehaviorAtom.
RT_TimeColorBehavior	0xF135	Specifies a TimeColorBehaviorAtom.

Name	Value	Meaning
RT_TimeEffectBehavior	0xF136	Specifies a TimeEffectBehaviorAtom.
RT_TimeMotionBehavior	0xF137	Specifies a TimeMotionBehaviorAtom.
RT_TimeRotationBehavior	0xF138	Specifies a TimeRotationBehaviorAtom.
RT_TimeScaleBehavior	0xF139	Specifies a TimeScaleBehaviorAtom.
RT_TimeSetBehavior	0xF13A	Specifies a TimeSetBehaviorAtom.
RT_TimeCommandBehavior	0xF13B	Specifies a TimeCommandBehaviorAtom.
RT_TimeClientVisualElement	0xF13C	Specifies a **ClientVisualElementContainer** (section 2.8.44).
RT_TimePropertyList	0xF13D	Specifies a **TimePropertyList4TimeNodeContainer** (section 2.8.18) or TimePropertyList2.8.37TimeBehavior.
RT_TimeVariantList	0xF13E	Specifies a **TimeStringListContainer** (section 2.8.36).
RT_TimeAnimationValueList	0xF13F	Specifies a **TimeAnimationValueListContainer** (section 2.8.31).
RT_TimeIterateData	0xF140	Specifies a TimeIterateDataAtom.
RT_TimeSequenceData	0xF141	Specifies a TimeSequenceDataAtom.
RT_TimeVariant	0xF142	Specifies a TimeVariantBool, TimeVariantInt, TimeVariantFloat, TimeVariantString, TimeDisplayType, TimeMasterRelType, TimeSubType, TimeEffectID, TimeEffectType, TimeNodeTimeFilter, TimeEventFilter, TimeGroupID, TimeEffectNodeType, TimeColorModel, TimeColorDirection, TimeOverride, TimeRuntimeContext, or TimePointsTypes.
RT_TimeAnimationValue	0xF143	Specifies a TimeAnimationValueAtom.
RT_TimeExtTimeNodeContainer	0xF144	Specifies an **ExtTimeNodeContainer** (section 2.8.15).
RT_TimeSubEffectContainer	0xF145	Specifies a SubEffectContainer.

2.13.25 SlideLayoutType

Referenced by: SlideAtom

An enumeration that specifies the slide layout of a slide.

Name	Value	Meaning
SL_TitleSlide	0x00000000	One title and one subtitle placeholder shapes.
SL_TitleBody	0x00000001	Presentation slide or main master slide layout with one title and one body placeholder shape.
SL_MasterTitle	0x00000002	Title master slide layout with one title and one subtitle placeholder shape.

Name	Value	Meaning
SL_TitleOnly	0x00000007	Presentation slide layout with one title placeholder shape.
SL_TwoColumns	0x00000008	Presentation slide layout with one title and two body placeholder shapes stacked horizontally.
SL_TwoRows	0x00000009	Presentation slide layout with one title and two body placeholder shapes stacked vertically.
SL_ColumnTwoRows	0x0000000A	Presentation slide layout with one title and three body placeholder shapes split into two columns. The right column has two rows.
SL_TwoRowsColumn	0x0000000B	Presentation slide layout with one title and three body placeholder shapes split into two columns. The left column has two rows.
SL_TwoColumnsRow	0x0000000D	Presentation slide layout with one title and three body placeholder shapes split into two rows. The top row has two columns.
SL_FourObjects	0x0000000E	Presentation slide layout with one title and four body placeholder shapes.
SL_BigObject	0x0000000F	Presentation slide layout with one body placeholder shape.
SL_Blank	0x00000010	Presentation slide layout with no placeholder shape.
SL_VerticalTitleBody	0x00000011	Presentation slide layout with a vertical title placeholder shape on the right and a body placeholder shape on the left.
SL_VerticalTwoRows	0x00000012	Presentation slide layout with a vertical title placeholder shape on the right and two body placeholder shapes in two columns on the left.

2.13.26 SlideSizeEnum

Referenced by: DocumentAtom

An enumeration that specifies types of slide sizes.

Name	Value	Meaning
SS_Screen	0x0000	Slide size ratio is consistent with a computer screen.
SS_LetterPaper	0x0001	Slide size ratio is consistent with letter paper.
SS_A4Paper	0x0002	Slide size ratio is consistent with A4 paper.
SS_35mm	0x0003	Slide size ratio is consistent with 35mm photo slides.
SS_Overhead	0x0004	Slide size ratio is consistent with overhead projector slides.
SS_Banner	0x0005	Slide size ratio is consistent with a banner.
SS_Custom	0x0006	Slide size ratio that is not consistent with any of the other specified slide sizes in this enumeration.

2.13.27 TextAlignmentEnum

Referenced by: TextPFException

An enumeration that specifies paragraph alignments.

Name	Value	Meaning
Tx_ALIGNLeft	0x0000	For horizontal text, left aligned. For vertical text, top aligned.
Tx_ALIGNCenter	0x0001	For horizontal text, centered. For vertical text, middle aligned.
Tx_ALIGNRight	0x0002	For horizontal text, right aligned. For vertical text, bottom aligned.
Tx_ALIGNJustify	0x0003	For horizontal text, flush left and right. For vertical text, flush top and bottom.
Tx_ALIGNDistributed	0x0004	Distribute space between characters.
Tx_ALIGNThaiDistributed	0x0005	Thai distribution justification.
Tx_ALIGNJustifyLow	0x0006	Kashida justify low.

2.13.28 TextAutoNumberSchemeEnum

Referenced by: TextAutoNumberScheme

An enumeration that specifies the character sequence and delimiters to use for automatic numbering.

Name	Value	Meaning
ANM_AlphaLcPeriod	0x0000	Lowercase Latin character followed by a period. Example: a., b., c., …
ANM_AlphaUcPeriod	0x0001	Uppercase Latin character followed by a period. Example: A., B., C., …
ANM_ArabicParenRight	0x0002	Arabic numeral followed by a closing parenthesis. Example: 1), 2), 3), …
ANM_ArabicPeriod	0x0003	Arabic numeral followed by a period. Example: 1., 2., 3., …
ANM_RomanLcParenBoth	0x0004	Lowercase Roman numeral enclosed in parentheses. Example: (i), (ii), (iii), …
ANM_RomanLcParenRight	0x0005	Lowercase Roman numeral followed by a closing parenthesis. Example: i), ii), iii), …
ANM_RomanLcPeriod	0x0006	Lowercase Roman numeral followed by a period. Example: i., ii., iii., …
ANM_RomanUcPeriod	0x0007	Uppercase Roman numeral followed by a period. Example: I., II., III., …
ANM_AlphaLcParenBoth	0x0008	Lowercase alphabetic character enclosed in parentheses. Example: (a), (b), (c), …
ANM_AlphaLcParenRight	0x0009	Lowercase alphabetic character followed by a closing parenthesis. Example: a), b), c), …

Name	Value	Meaning
ANM_AlphaUcParenBoth	0x000A	Uppercase alphabetic character enclosed in parentheses. Example: (A), (B), (C), …
ANM_AlphaUcParenRight	0x000B	Uppercase alphabetic character followed by a closing parenthesis. Example: A), B), C), …
ANM_ArabicParenBoth	0x000C	Arabic numeral enclosed in parentheses. Example: (1), (2), (3), …
ANM_ArabicPlain	0x000D	Arabic numeral. Example: 1, 2, 3, …
ANM_RomanUcParenBoth	0x000E	Uppercase Roman numeral enclosed in parentheses. Example: (I), (II), (III), …
ANM_RomanUcParenRight	0x000F	Uppercase Roman numeral followed by a closing parenthesis. Example: I), II), III), …
ANM_ChsPlain	0x0010	Simplified Chinese.
ANM_ChsPeriod	0x0011	Simplified Chinese with single-byte period.
ANM_CircleNumDBPlain	0x0012	Double byte circle numbers.
ANM_CircleNumWDBWhitePlain	0x0013	Wingdings white circle numbers.
ANM_CircleNumWDBBlackPlain	0x0014	Wingdings black circle numbers.
ANM_ChtPlain	0x0015	Traditional Chinese.
ANM_ChtPeriod	0x0016	Traditional Chinese with single-byte period.
ANM_Arabic1Minus	0x0017	Bidi Arabic 1 (AraAlpha) with ANSI minus symbol.
ANM_Arabic2Minus	0x0018	Bidi Arabic 2 (AraAbjad) with ANSI minus symbol.
ANM_Hebrew2Minus	0x0019	Bidi Hebrew 2 with ANSI minus symbol.
ANM_JpnKorPlain	0x001A	Japanese/Korean.
ANM_JpnKorPeriod	0x001B	Japanese/Korean with single-byte period.
ANM_ArabicDbPlain	0x001C	Double-byte Arabic numbers.
ANM_ArabicDbPeriod	0x001D	Double-byte Arabic numbers with double-byte period.
ANM_ThaiAlphaPeriod	0x001E	Thai alphabetic character followed by a period.
ANM_ThaiAlphaParenRight	0x001F	Thai alphabetic character followed by a closing parenthesis.
ANM_ThaiAlphaParenBoth	0x0020	Thai alphabetic character enclosed by parentheses.
ANM_ThaiNumPeriod	0x0021	Thai numeral followed by a period.
ANM_ThaiNumParenRight	0x0022	Thai numeral followed by a closing parenthesis.
ANM_ThaiNumParenBoth	0x0023	Thai numeral enclosed in parentheses.
ANM_HindiAlphaPeriod	0x0024	Hindi alphabetic character followed by a period.

Name	Value	Meaning
ANM_HindiNumPeriod	0x0025	Hindi numeric character followed by a period.
ANM_JpnChsDBPeriod	0x0026	Japanese with double-byte period.
ANM_HindiNumParenRight	0x0027	Hindi numeric character followed by a closing parenthesis.
ANM_HindiAlpha1Period	0x0028	Hindi alphabetic character followed by a period.

2.13.29 TextBuildSubEffectEnum

Referenced by: AnimationInfoAtom

An enumeration that specifies behavior types of text in animation effects.

Name	Value	Meaning
TXB_BuildByNone	0x00	Text is animated all at once.
TXB_BuildByWord	0x01	Text is animated word by word.
TXB_BuildByCharacter	0x02	Text is animated character by character.

2.13.30 TextDirectionEnum

Referenced by: TextPFException

An enumeration that specifies the direction of a paragraph of text.

Name	Value	Meaning
LeftToRight	0x0000	Left to right text flow.
RightToLeft	0x0001	Right to left text flow.

2.13.31 TextFontAlignmentEnum

Referenced by: TextPFException

An enumeration that specifies font alignment.

Name	Value	Meaning
Tx_ALIGNFONTRoman	0x0000	Place characters on font baseline.
Tx_ALIGNFONTHanging	0x0001	Characters hang from top of line height
Tx_ALIGNFONTCenter	0x0002	Characters centered within line height.
Tx_ALIGNFONTUpholdFixed	0x0003	Characters are anchored to the very bottom of a single line. This is different than Tx_ALIGNFONTRoman because of letters such as "g", "q", and "y".

2.13.32 TextTabTypeEnum

Referenced by: TabStop

An enumeration that specifies alignment types of tab stops.

Name	Value	Meaning
Tx_TABLeft	0x0000	Left-aligned tab stop.
Tx_TABCenter	0x0001	Center-aligned tab stop.
Tx_TABRight	0x0002	Right-aligned tab stop.
Tx_TABDecimal	0x0003	Decimal point-aligned tab stop.

2.13.33 TextTypeEnum

Referenced by: OutlineTextPropsHeaderExAtom, TextHeaderAtom

An enumeration that specifies the types of text.

Name	Value	Meaning
Tx_TYPE_TITLE	0x00000000	Title placeholder shape text.
Tx_TYPE_BODY	0x00000001	Body placeholder shape text.
Tx_TYPE_NOTES	0x00000002	Notes placeholder shape text.
Tx_TYPE_OTHER	0x00000004	Any other text.
Tx_TYPE_CENTERBODY	0x00000005	Center body placeholder shape text.
Tx_TYPE_CENTERTITLE	0x00000006	Center title placeholder shape text.
Tx_TYPE_HALFBODY	0x00000007	Half-sized body placeholder shape text.
Tx_TYPE_QUARTERBODY	0x00000008	Quarter-sized body placeholder shape text.

2.13.34 TimeAnimateBehaviorValueTypeEnum

Referenced by: TimeAnimateBehaviorAtom, TimeSetBehaviorAtom

An enumeration that specifies the data type of a property to be animated.

Name	Value	Meaning
TL_TABVT_String	0x00000000	Animate text content.
TL_TABVT_Number	0x00000001	Animate a numeric property.
TL_TABVT_Color	0x00000002	Animate a color property.

2.13.35 TimeCommandBehaviorTypeEnum

Referenced by: TimeCommandBehaviorAtom

An enumeration that specifies the type of a command.

Name	Value	Meaning
TL_TCBT_Event	0x00000000	Send out an event to the target object.

Name	Value	Meaning
TL_TCBT_Call	0x00000001	Call a method or function on the target object.
TL_TCBT_OleVerb	0x00000002	Send an OLE verb to the target object.

2.13.36 TimeNodeTypeEnum

Referenced by: TimeNodeAtom

An enumeration that specifies the type of a time node.

Name	Value	Meaning
TL_TNT_Parallel	0x00000000	Parallel time node whose child nodes can start simultaneously.
TL_TNT_Sequential	0x00000001	Sequential time node whose child nodes can only start sequentially and each child can only start after its previous sibling has started.
TL_TNT_Behavior	0x00000003	Behavior time node that contains a behavior.
TL_TNT_Media	0x00000004	Media time node that contains a media object.

2.13.37 TimePropertyID4TimeBehavior

Referenced by: TimeVariant2.8.38Behavior

An enumeration that specifies the type of attributes for an animation behavior.

Name	Value	Meaning
TL_TBPID_UnknownPropertyList	0x00000001	Unknown property list.
TL_TBPID_RuntimeContext	0x00000002	Runtime context that specifies which versions of the application can run the behavior.
TL_TBPID_MotionPathEditRelative	0x00000003	Whether a motion path moves with the object that it applies to during editing.
TL_TBPID_ColorColorModel	0x00000004	Color model of a color animation.
TL_TBPID_ColorDirection	0x00000005	Color direction of a color animation.
TL_TBPID_Override	0x00000006	How to override animated values.
TL_TBPID_PathEditRotationAngle	0x00000007	Rotation angle of a motion path.
TL_TBPID_PathEditRotationX	0x00000008	Horizontal position of the rotation center of the motion path.
TL_TBPID_PathEditRotationY	0x00000009	Vertical position of the rotation center of the motion path.
TL_TBPID_PointsTypes	0x0000000A	The type of points in the motion path.

2.13.38 TimePropertyID4TimeNode

Referenced by: TimeVariant2.8.19TimeNode

An enumeration that specifies the type of attributes for a time node.

Name	Value	Meaning
TL_TPID_Display	0x00000002	Display type in UI.
TL_TPID_MasterPos	0x00000005	Relationship to the master time node.
TL_TPID_SubType	0x00000006	Type of the subordinate time node.
TL_TPID_EffectID	0x00000009	Identifier of an animation effect.
TL_TPID_EffectDir	0x0000000A	Direction of an animation effect.
TL_TPID_EffectType	0x0000000B	Type of an animation effect.
TL_TPID_AfterEffect	0x0000000D	Whether the time node is an after effect.
TL_TPID_SlideCount	0x0000000F	The number of slides that a media will play across.
TL_TPID_TimeFilter	0x00000010	Time filtering for the time node.
TL_TPID_EventFilter	0x00000011	Event filtering for the time node.
TL_TPID_HideWhenStopped	0x00000012	Whether to display the media when it is stopped.
TL_TPID_GroupID	0x00000013	Build identifier.
TL_TPID_EffectNodeType	0x00000014	The role of the time node in the timing structure.
TL_TPID_PlaceholderNode	0x00000015	Whether the time node is a placeholder.
TL_TPID_MediaVolume	0x00000016	The volume of a media.
TL_TPID_MediaMute	0x00000017	Whether a media object is mute.
TL_TPID_ZoomToFullScreen	0x0000001A	Whether to zoom a media object to full screen.

2.13.39 TimeVariantTypeEnum

Referenced by: TimeColorDirection, TimeColorModel, TimeDisplayType, TimeEffectID, TimeEffectNodeType, TimeEffectType, TimeEventFilter, TimeGroupID, TimeMasterRelType, TimeNodeTimeFilter, TimeOverride, TimePointsTypes, TimeRuntimeContext, TimeSubType, TimeVariantBool, TimeVariantFloat, TimeVariantInt, TimeVariantString

An enumeration that specifies the data type of a value.

Name	Value	Meaning
TL_TVT_Bool	0x00	A **Boolean** value.
TL_TVT_Int	0x01	A **signed integer**.
TL_TVT_Float	0x02	A floating-point number.
TL_TVT_String	0x03	A **Unicode** string.

2.13.40 TimeVisualElementEnum

Referenced by: VisualPageAtom, VisualShapeChartElementAtom, VisualShapeGeneralAtom, VisualSoundAtom

An enumeration that specifies the part of a slide or shape to which the animation is applied.

Name	Value	Meaning
TL_TVET_Shape	0x00000000	Applies to the shape and all its text.
TL_TVET_Page	0x00000001	Applies to the slide.
TL_TVET_TextRange	0x00000002	Applies to a specified range of text of the shape.
TL_TVET_Audio	0x00000003	Applies to the audio of the shape.
TL_TVET_Video	0x00000004	Applies to the video of the shape.
TL_TVET_ChartElement	0x00000005	Applies to the elements of the chart.
TL_TVET_ShapeOnly	0x00000006	Applies to the shape but not its text.
TL_TVET_AllTextRange	0x00000008	Applies to all text of the shape.

2.13.41 TriggerObjectEnum

Referenced by: TimeConditionAtom

An enumeration that specifies the type of a target that participates in the evaluation of a time condition.

Name	Value	Meaning
TL_TOT_None	0x00000000	None.
TL_TOT_VisualElement	0x00000001	An animatable object.
TL_TOT_TimeNode	0x00000002	A time node.
TL_TOT_RuntimeNodeRef	0x00000003	Runtime child time nodes.

2.13.42 ViewTypeEnum

Referenced by: UserEditAtom

An enumeration that specifies different viewing modes in which a presentation can be displayed in a user interface.

Name	Value	Meaning
V_Slide	0x0001	A view optimized for the display of a presentation slide.
V_SlideMaster	0x0002	A view optimized for the display of a main master slide.
V_Notes	0x0003	A view optimized for the display of a notes slide.
V_Handout	0x0004	A view optimized for the display of the handout master slide.
V_NotesMaster	0x0005	A view optimized for the display of the notes master slide.
V_OutlineMaster	0x0006	A view optimized for the display of the outline master slide.
V_Outline	0x0007	A view optimized for the display of the text on the presentation slides.
V_SlideSorter	0x0008	A view optimized for the simultaneous display of multiple presentation slides.

Name	Value	Meaning
V_VisualBasic	0x0009	A view optimized for the display of the VBA information.
V_TitleMaster	0x000A	A view optimized for the display of a title master slide.
V_SlideShow	0x000B	A view optimized for the display of a slide show.
V_SlideShowFullScreen	0x000C	A view optimized for the display of a slide show in full screen.
V_NotesText	0x000D	A view optimized for the display of the text of a notes slide.
V_PrintPreview	0x000E	A view optimized for the display of a print preview of the presentation slides.
V_Thumbnails	0x000F	A view optimized for the simultaneous display of multiple presentation slides in a single column.
V_MasterThumbnails	0x0010	A view optimized for the simultaneous display of multiple main master slides or title master slides in a single column.
V_PodiumSlideView	0x0011	A view optimized for the display of presentation slides while a slide show is also being displayed.
V_PodiumNotesView	0x0012	A view optimized for the display of the text of a notes slide while a slide show is also being displayed.

2.13.43 WebFrameColorsEnum

Referenced by: HTMLDocInfo9Atom

An enumeration that specifies the color options for displaying the text and background for the Web page notes pane and outline pane.

Name	Value	Meaning
MSOWOPTBrowserColors	0x0000	Browser colors.
MSOWOPTPresentationSchemeTextColor	0x0001	Presentation text colors.
MSOWOPTPresentationSchemeAccentColor	0x0002	Presentation accent colors.
MSOWOPTWhiteTextOnBlack	0x0003	White text on black background.
MSOWOPTBlackTextOnWhite	0x0004	Black text on white background.

2.13.44 WebOutputEnum

Referenced by: HTMLDocInfo9Atom, HTMLPublishInfoAtom

An enumeration that specifies the target Web technology support for which a publishing is optimized.

Name	Value	Meaning
HTML_EXPORTVersion3	0x01	Web page is optimized for use with HTML, CSS, Javascript and frames.
HTML_EXPORTVersion4	0x02	Web page is optimized for use with HTML, MHTML, DHTML, CSS, JScript, frames, VML, and PNG graphics.
HTML_EXPORTDual	0x04	Web page contains optimized output for both technology formats and conditional statements to determine proper usage.

3 Structure Examples

3.1 Introduction

The following sections provide structure examples of different features of this file format. These examples are illustrative of the preceding specification and do not cover all possible structure usage scenarios. They are not intended to replace the preceding specification but rather to clarify and enhance it. All examples are derived from the same presentation, sample.ppt, which is represented visually in the following figures.

Figure 4: Outline text

Figure 5: Presentation slide 1

Figure 6: Presentation slide 2

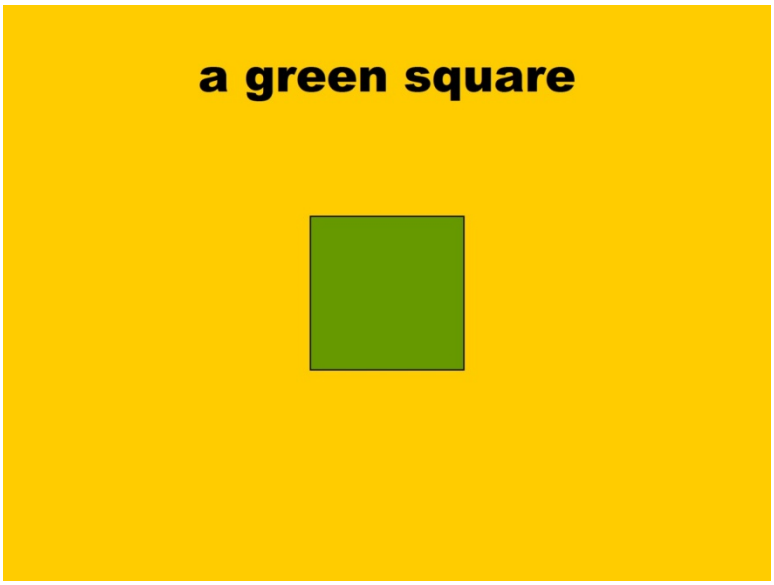


Figure 7: Presentation slide 3

Figure 8: Presentation slide 4

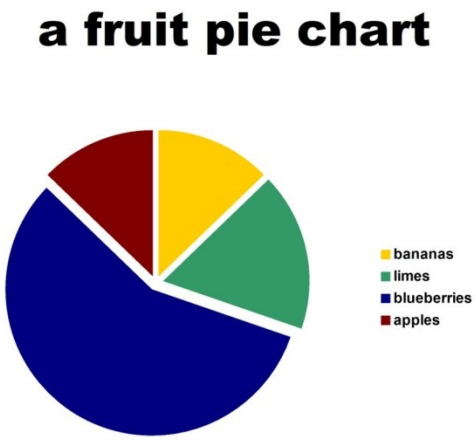


Figure 9: Presentation slide 5

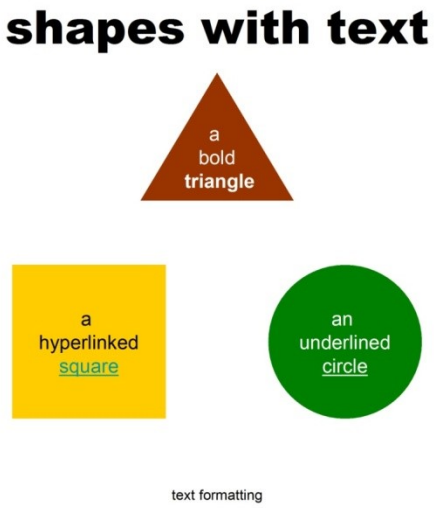


Figure 10: Presentation slide 6

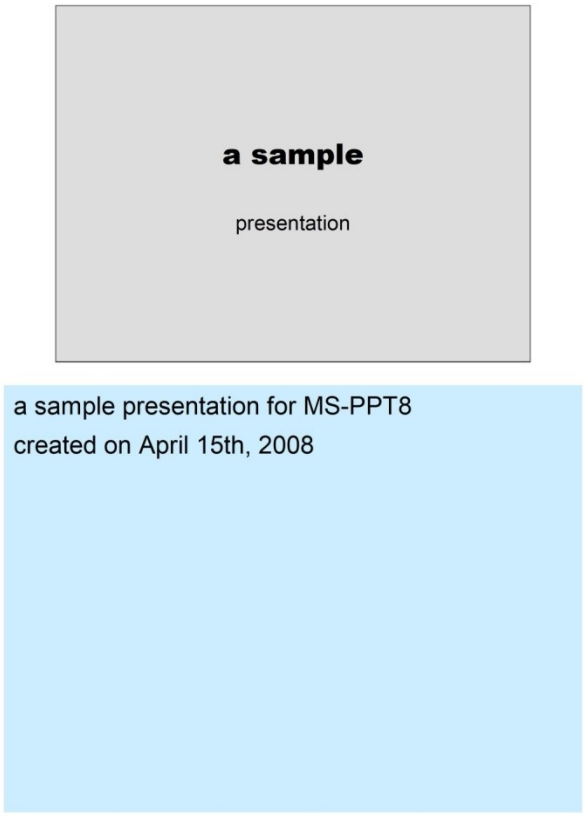


Figure 11: Notes slide 1

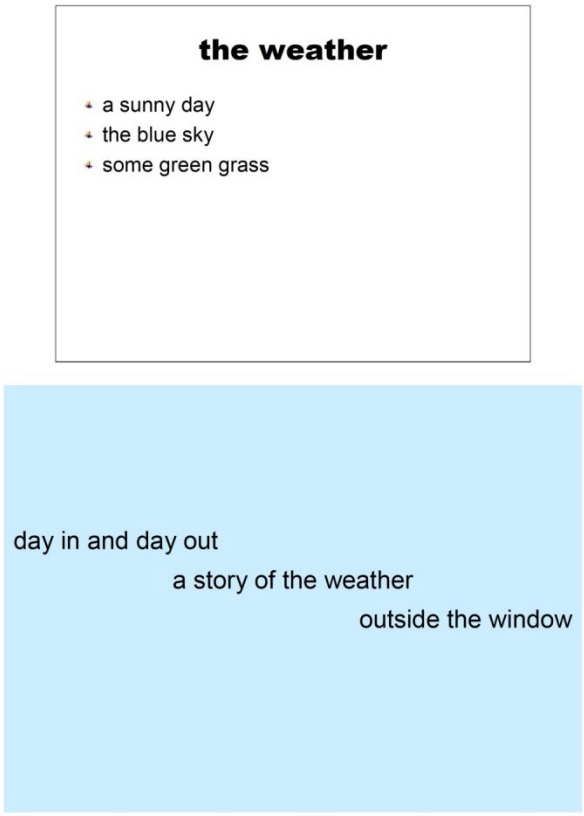


Figure 12: Notes slide 2

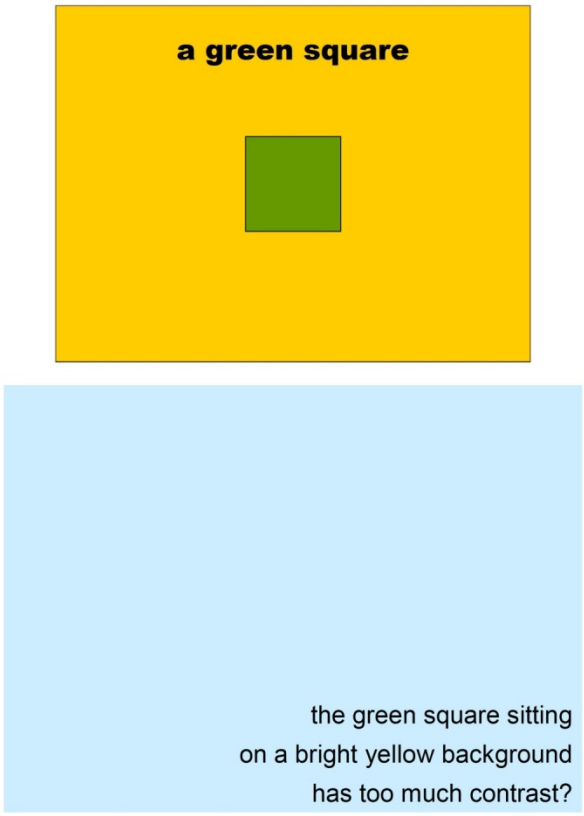


Figure 13: Notes slide 3

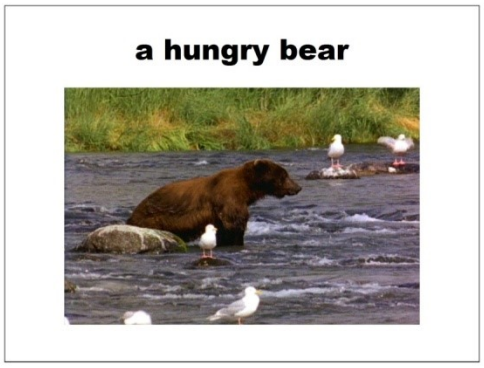
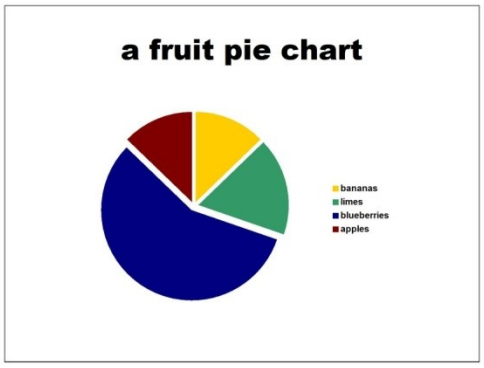


Figure 14: Notes slide 4



in all, how much fruit
can someone possibly eat
a whole pie, i guess

5

Figure 15: Notes slide 5

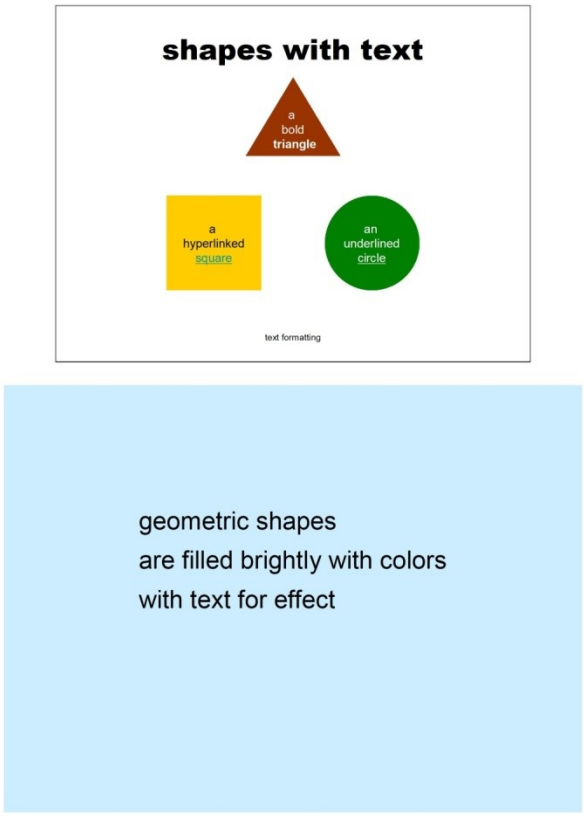


Figure 16: Notes slide 6

Figure 17: Main master slide

Figure 18: Title master slide

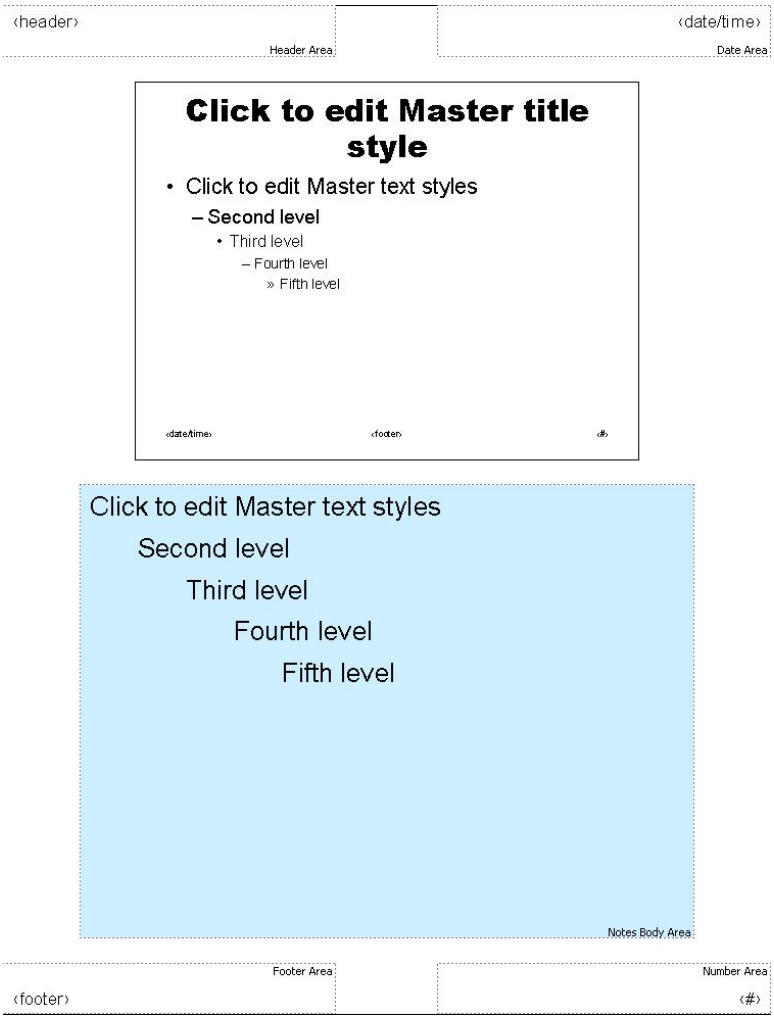


Figure 19: Notes master slide

The sample presentation was authored in two sessions. The first session created a main master slide, a title master slide, six presentation slides, and associated notes slides for five of the six presentation slides. The second session deleted the presentation slide without an associated notes slide, changed the content on presentation slide 3, and added presentation slide 6 and an associated notes slide.

The preceding figures will be further explained in the sections that follow.

3.2 File Structure Example

This section provides an example of how to construct the persist object directory for the file. Correctly constructing the persist object directory is the first step in finding the rest of the presentation content in the file—main master slides, notes master slide, handout master slide, presentation slides, notes slides, embedded or linked OLE object storages, and the VBA project storage. Because new user edits can simply be appended to the existing stream contents, the process for constructing the persist object directory needs to be carefully followed to ensure that the most recent version of a persist object is used and older versions are ignored.

Construction of the persist object directory begins with parsing the **CurrentUserAtom** record (section 2.3.2). The child-record hierarchy of the **CurrentUserAtom** record is shown expanded in the following table.

Offset	Size	Structure	Value
00000000	005F	CurrentUserAtom **- currentUserAtom**	
00000000	0008	RecordHeader **- rh**	
00000008	0004	unsigned integer **- size**	0x00000014
0000000C	0004	unsigned integer **- headerToken**	0xE391C05F
00000010	0004	unsigned integer **- offsetToCurrentEdit**	0x00008290
00000014	0002	unsigned integer **- lenUserName**	0x0015
00000016	0002	unsigned integer **- docFileVersion**	0x03F4
00000018	0001	unsigned integer **- majorVersion**	0x03
00000019	0001	unsigned integer **- minorVersion**	0x00
0000001A	0002	unsigned integer **- unused**	0x0000
0000001C	0015	array of bytes **- ansiUserName**	Microsoft Corporation
00000031	0004	unsigned integer **- relVersion**	0x00000008
00000035	002A	array of bytes **- unicodeUserName**	Microsoft Corporation

Figure 20: CurrentUserAtom child-record hierarchy

offsetToCurrentEdit: 0x00008290 specifies the offset, in bytes, from the beginning of the **PowerPoint Document Stream** (section 2.1.2) to the **UserEditAtom** record (section 2.3.3) for the most recent user edit.

The **UserEditAtom** record specified by the **offsetToCurrentEdit** field in the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00008290	0024	UserEditAtom	
00008290	0008	RecordHeader **- rh**	
00008298	0004	unsigned integer **- lastSlideIdRef**	0x00000100
0000829C	16 bits	unsigned integer **- version**	0x1FE9
0000829C	8 bits	unsigned integer **- minorVersion**	0x00
0000829C	8 bits	unsigned integer **- majorVersion**	0x03
000082A0	0004	unsigned integer **- offsetLastEdit**	0x00005C45
000082A4	0004	unsigned integer **- offsetPersistDirectory**	0x0000826C
000082A8	0004	unsigned integer **- docPersistIdRef**	0x00000001
000082AC	0004	unsigned integer **- persistIdSeed**	0x00000014
000082B0	0002	ViewTypeEnum **- lastView**	0x0001
000082B2	0002	unsigned integer **- unused**	0x31C5

Figure 21: UserEditAtom child-record hierarchy for the most recent user edit

offsetLastEdit: 0x00005C45 specifies the offset, in bytes, from the beginning of the **PowerPoint Document Stream** to the **UserEditAtom** record for the previous user edit.

offsetPersistDirectory: 0x0000826C specifies the offset, in bytes, from the beginning of the **PowerPoint Document Stream** to the **PersistDirectoryAtom** record (section 2.3.4) for the most recent user edit.

The **PersistDirectoryAtom** record specified by the **offsetPersistDirectory** field in the previous table titled "UserEditAtom child-record hierarchy for the most recent user edit" is shown expanded in the following table.

Offset	Size	Structure	Value
0000826C	0024	PersistDirectoryAtom	
0000826C	0008	RecordHeader **- rh**	
00008274	0008	PersistDirectoryEntry **- persistDirEntry[0]**	
00008274	20 bits	unsigned integer **- persistId**	0x00001
00008274	12 bits	unsigned integer **- cPersist**	0x001
00008278	0004	PersistOffsetEntry **- persistOffset[0]**	0x00005C69
0000827C	0008	PersistDirectoryEntry **- persistDirEntry[1]**	
0000827C	20 bits	unsigned integer **- persistId**	0x00006
0000827C	12 bits	unsigned integer **- cPersist**	0x001
00008280	0004	PersistOffsetEntry **- persistOffset[0]**	0x00006964
00008284	000C	PersistDirectoryEntry **- persistDirEntry[2]**	
00008284	20 bits	unsigned integer **- persistId**	0x00013
00008284	12 bits	unsigned integer **- cPersist**	0x002
00008288	0004	PersistOffsetEntry **- persistOffset[0]**	0x00007AF6
0000828C	0004	PersistOffsetEntry **- persistOffset[1]**	0x00007FE3

Figure 22: PersistDirectoryAtom child-record hierarchy for the most recent user edit

persistDirEntry[0].persistId: 0x00001 specifies the starting persist object identifier for the **PersistOffsetEntry** entries (section 2.3.6) that follow.

persistDirEntry[0].cPersist: 0x001 specifies the count of **PersistOffsetEntry** entries that follow.

persistDirEntry[0].persistOffset[0]: 0x00005C69 specifies the stream offset for the persist object with persist object identifier 0x00001.

persistDirEntry[1].persistId: 0x00006 specifies the starting persist object identifier for the **PersistOffsetEntry** entries that follow.

persistDirEntry[1].cPersist: 0x001 specifies the count of **PersistOffsetEntry** entries that follow.

persistDirEntry[1].persistOffset[0]: 0x00006964 specifies the stream offset for the persist object with persist object identifier 0x00006.

persistDirEntry[2].persistId: 0x00013 specifies the starting persist object identifier for the **PersistOffsetEntry** entries that follow.

persistDirEntry[2].cPersist: 0x002 specifies the count of **PersistOffsetEntry** entries that follow.

persistDirEntry[2].persistOffset[0]: 0x00007AF6 specifies the stream offset for the persist object with persist object identifier 0x00013.

persistDirEntry[2].persistOffset[1]: 0x00007FE3 specifies the stream offset for the persist object with persist object identifier 0x00014.

The partial persist object directory constructed after parsing the first **PersistDirectoryAtom** record is summarized in the following table.

Persist object identifier	Persist object stream offset
0x00001	0x00005C69
0x00006	0x00006964
0x00013	0x00007AF6
0x00014	0x00007FE3

The **UserEditAtom** record specified by the **offsetLastEdit** field in the previous table titled "UserEditAtom child-record hierarchy for the most recent user edit" is shown expanded in the following table.

Offset	Size	Structure	Value
00005C45	0024	UserEditAtom	
00005C45	0008	RecordHeader **- rh**	
00005C4D	0004	unsigned integer **- lastSlideIdRef**	0x00000104
00005C51	16 bits	unsigned integer **- version**	0x1FE9
00005C51	8 bits	unsigned integer **- minorVersion**	0x00
00005C51	8 bits	unsigned integer **- majorVersion**	0x03
00005C55	0004	unsigned integer **- offsetLastEdit**	0x00000000
00005C59	0004	unsigned integer **- offsetPersistDirectory**	0x00005BF1
00005C5D	0004	unsigned integer **- docPersistIdRef**	0x00000001
00005C61	0004	unsigned integer **- persistIdSeed**	0x00000012
00005C65	0002	ViewTypeEnum **- lastView**	0x000F
00005C67	0002	unsigned integer **- unused**	0x31C5

Figure 23: UserEditAtom child-record hierarchy for the previous user edit

offsetLastEdit: 0x00000000 specifies that no further user edits exist.

offsetPersistDirectory: 0x00005BF1 specifies the offset, in bytes, from the beginning of the **PowerPoint Document Stream** to the **PersistDirectoryAtom** record for this previous user edit.

The **PersistDirectoryAtom** record specified by the **offsetPersistDirectory** field in the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00005BF1	0054	PersistDirectoryAtom	
00005BF1	0008	RecordHeader **- rh**	
00005BF9	0028	PersistDirectoryEntry **- persistDirEntry[0]**	
00005BF9	20 bits	unsigned integer **- persistId**	0x00001
00005BF9	12 bits	unsigned integer **- cPersist**	0x009
00005BFD	0004	PersistOffsetEntry **- persistOffset[0]**	0x00000000
00005C01	0004	PersistOffsetEntry **- persistOffset[1]**	0x00000CC5
00005C05	0004	PersistOffsetEntry **- persistOffset[2]**	0x00001671
00005C09	0004	PersistOffsetEntry **- persistOffset[3]**	0x000021C8
00005C0D	0004	PersistOffsetEntry **- persistOffset[4]**	0x000023F4
00005C11	0004	PersistOffsetEntry **- persistOffset[5]**	0x000030CD
00005C15	0004	PersistOffsetEntry **- persistOffset[6]**	0x00004D69
00005C19	0004	PersistOffsetEntry **- persistOffset[7]**	0x000037B5
00005C1D	0004	PersistOffsetEntry **- persistOffset[8]**	0x00003E83
00005C21	0008	PersistDirectoryEntry **- persistDirEntry[1]**	
00005C21	20 bits	unsigned integer **- persistId**	0x0000B
00005C21	12 bits	unsigned integer **- cPersist**	0x001
00005C25	0004	PersistOffsetEntry **- persistOffset[0]**	0x00001B98
00005C29	001C	PersistDirectoryEntry **- persistDirEntry[2]**	
00005C29	20 bits	unsigned integer **- persistId**	0x0000D
00005C29	12 bits	unsigned integer **- cPersist**	0x006
00005C2D	0004	PersistOffsetEntry **- persistOffset[0]**	0x000040CB
00005C31	0004	PersistOffsetEntry **- persistOffset[1]**	0x00004319
00005C35	0004	PersistOffsetEntry **- persistOffset[2]**	0x000045D4
00005C39	0004	PersistOffsetEntry **- persistOffset[3]**	0x00004873
00005C3D	0004	PersistOffsetEntry **- persistOffset[4]**	0x00004B22
00005C41	0004	PersistOffsetEntry **- persistOffset[5]**	0x00002E81

Figure 24: PersistDirectoryAtom child-record hierarchy for the previous user edit

persistDirEntry[0].persistId: 0x00001 specifies the starting persist object identifier for the **PersistOffsetEntry** entries that follow.

persistDirEntry[0].cPersist: 0x009 specifies the count of **PersistOffsetEntry** entries that follow.

persistDirEntry[0].persistOffset[0]: 0x00000000 specifies the stream offset for the persist object with persist object identifier 0x00001. However, because the persist object identifier 0x00001 already appears in the first table in this section, this stream offset is ignored.

persistDirEntry[0].persistOffset[1]: 0x00000CC5 specifies the stream offset for the persist object with persist object identifier 0x00002.

persistDirEntry[0].persistOffset[2]: 0x00001671 specifies the stream offset for the persist object with persist object identifier 0x00003.

persistDirEntry[0].persistOffset[3]: 0x000021C8 specifies the stream offset for the persist object with persist object identifier 0x00004.

persistDirEntry[0].persistOffset[4]: 0x000023F4 specifies the stream offset for the persist object with persist object identifier 0x00005.

persistDirEntry[0].persistOffset[5]: 0x000030CD specifies the stream offset for the persist object with persist object identifier 0x00006. However, because the persist object identifier 0x00006 already appears in the first table in this section, this stream offset is ignored.

persistDirEntry[0].persistOffset[6]: 0x00004D69 specifies the stream offset for the persist object with persist object identifier 0x00007.

persistDirEntry[0].persistOffset[7]: 0x000037B5 specifies the stream offset for the persist object with persist object identifier 0x00008.

persistDirEntry[0].persistOffset[8]: 0x00003E83 specifies the stream offset for the persist object with persist object identifier 0x00009.

persistDirEntry[1].persistId: 0x0000B specifies the starting persist object identifier for the **PersistOffsetEntry** entries that follow.

persistDirEntry[1].cPersist: 0x001 specifies the count of **PersistOffsetEntry** entries that follow.

persistDirEntry[1].persistOffset[0]: 0x00001B98 specifies the stream offset for the persist object with persist object identifier 0x0000B.

persistDirEntry[2].persistId: 0x0000D specifies the starting persist object identifier for the **PersistOffsetEntry** entries that follow.

persistDirEntry[2].cPersist: 0x006 specifies the count of **PersistOffsetEntry** entries that follow.

persistDirEntry[2].persistOffset[0]: 0x000040CB specifies the stream offset for the persist object with persist object identifier 0x0000D.

persistDirEntry[2].persistOffset[1]: 0x00004319 specifies the stream offset for the persist object with persist object identifier 0x0000E.

persistDirEntry[2].persistOffset[2]: 0x000045D4 specifies the stream offset for the persist object with persist object identifier 0x0000F.

persistDirEntry[2].persistOffset[3]: 0x00004873 specifies the stream offset for the persist object with persist object identifier 0x00010.

persistDirEntry[2].persistOffset[4]: 0x00004B22 specifies the stream offset for the persist object with persist object identifier 0x00011.

persistDirEntry[2].persistOffset[5]: 0x00002E81 specifies the stream offset for the persist object with persist object identifier 0x00012.

The complete persist object directory constructed after parsing all **PersistDirectoryAtom** records is summarized in the following table.

Persist object identifier	Persist object stream offset
0x00001	0x00005C69
0x00002	0x00000CC5
0x00003	0x00001671
0x00004	0x000021C8
0x00005	0x000023F4
0x00006	0x00006964
0x00007	0x00004D69
0x00008	0x000037B5
0x00009	0x00003E83
0x0000B	0x00001B98
0x0000D	0x000040CB
0x0000E	0x00004319
0x0000F	0x000045D4
0x00010	0x00004873
0x00011	0x00004B22
0x00012	0x00002E81
0x00013	0x00007AF6
0x00014	0x00007FE3

3.3 Persist Objects Example

The **PowerPoint Document Stream** (section 2.1.2) is fundamentally a sequence of container records and atom records that represent persist objects, the **PersistDirectoryAtom** records (section 2.3.4) that comprise the persist object directory, and the **UserEditAtom** records (section 2.3.3) that identify the content comprising each user edit. The specific top-level record sequence for the sample presentation is shown in the following table.

Offset	Size	Structure
00000000	82B4	Stream **- PowerPoint Document**
00000000	0CC5	**A**: DocumentContainer
00000CC5	09AC	**B**: MainMasterContainer
00001671	0527	**C**: SlideContainer
00001B98	0630	**D**: NotesContainer

Offset	Size	Structure
000021C8	022C	**E**: SlideContainer
000023F4	0A8D	**F**: SlideContainer
00002E81	024C	**G**: SlideContainer
000030CD	06E8	**H**: SlideContainer
000037B5	06CE	**I**: SlideContainer
00003E83	0248	**J**: SlideContainer
000040CB	024E	**K**: NotesContainer
00004319	02BB	**L**: NotesContainer
000045D4	029F	**M**: NotesContainer
00004873	02AF	**N**: NotesContainer
00004B22	0247	**O**: NotesContainer
00004D69	0E88	**P**: ExOleObjStg
00005BF1	0054	**Q**: PersistDirectoryAtom
00005C45	0024	**R**: UserEditAtom
00005C69	0CFB	**S**: DocumentContainer
00006964	1192	**T**: SlideContainer
00007AF6	04ED	**U**: SlideContainer
00007FE3	0289	**V**: NotesContainer
0000826C	0024	**W**: PersistDirectoryAtom
00008290	0024	**X**: UserEditAtom

Figure 25: Top-level record sequence in the PowerPoint Document Stream from sample.ppt

For each record in the previous table, the type of the record and its offset from the beginning of the **PowerPoint Document Stream** are listed. The atom records are: **ExOleObjStg** (section 2.10.34), **PersistDirectoryAtom** and **UserEditAtom**. The rest are container records. The letter labels are used to identify specific records in the text that follows.

The first user edit comprises records labeled A through R and the second user edit comprises records labeled S through X. By cross-referencing the previous table with the output from the second table in section 3.2, some dead records can be immediately identified, because their offsets do not appear in the second table in section 3.2. Specifically, **DocumentContainer** (section 2.4.1) record A is a dead record and is superseded by record S; **SlideContainer** (section 2.5.1) record H is also a dead record and is superseded by record T. Although **SlideContainer** record G is referenced by the persist object directory shown in the second table in section 3.2, further parsing of the **slideList** field of the **DocumentContainer** record S determines that no **SlidePersistAtom** (section 2.4.14.5) child-record has a **persistIdRef** field equal to 0x00000012. For more information about how the records from the previous table are associated with the slide images as shown in figures titled "Presentation slide 1" through "Notes master slide" in section 3.1, see the Slides Example (section 3.5).

3.4 Outline Text Example

The outline text can be displayed visually, as shown in the figure titled "Outline text" in section 3.1. The records used to construct the outline text are found inside the **SlideListWithTextContainer** record (section 2.4.14.3) contained within the **DocumentContainer** record (section 2.4.1). The child-record hierarchy of the **DocumentContainer** record S from the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3 is shown expanded in the following table.

Offset	Size	Structure
00005C69	0CFB	DocumentContainer
00005C69	0008	RecordHeader **- rh**
00005C71	0030	**E**: DocumentAtom **- documentAtom**
00005CA1	0152	**F**: ExObjListContainer **- exObjList**
00005DF3	016A	DocumentTextInfoContainer **- documentTextInfo**
00005F5D	01FC	DrawingGroupContainer **- drawingGroup**
00006159	0040	**A**: MasterListWithTextContainer **- masterList**
00006199	04D7	**D**: DocInfoListContainer **- docInfoList**
00006670	0014	SlideHeadersFootersContainer **- slideHF**
00006684	0014	NotesHeadersFootersContainer **- notesHF**
00006698	0214	**B**: SlideListWithTextContainer **- slideList**
000068AC	00B0	**C**: NotesListWithTextContainer **- notesList**
0000695C	0008	EndDocumentAtom **- endDocumentAtom**

Figure 26: DocumentContainer child-record hierarchy

The child-record hierarchy of the **SlideListWithTextContainer** (section 2.4.14.3) record B from the previous table is shown expanded in the following table.

Offset	Size	Structure
00006698	0214	SlideListWithTextContainer **- slideList**
00006698	0008	RecordHeader **- rh**
000066A0	001C	SlidePersistAtom **- case of RT_SlidePersistAtom**
000066BC	000C	**A**: TextHeaderAtom **- case of RT_TextHeaderAtom**
000066C8	0010	**B**: TextBytesAtom **- case of RT_TextBytesAtom**
000066D8	000C	**C**: TextHeaderAtom **- case of RT_TextHeaderAtom**
000066E4	0014	**D**: TextBytesAtom **- case of RT_TextBytesAtom**
000066F8	001C	SlidePersistAtom **- case of RT_SlidePersistAtom**
00006714	000C	**E**: TextHeaderAtom **- case of RT_TextHeaderAtom**
00006720	0013	**F**: TextBytesAtom **- case of RT_TextBytesAtom**

Offset	Size	Structure
00006733	000C	**G**: TextHeaderAtom **- case of RT_TextHeaderAtom**
0000673F	0031	**H**: TextBytesAtom **- case of RT_TextBytesAtom**
00006770	0022	StyleTextPropAtom **- case of RT_StyleTextPropAtom**
00006792	001C	SlidePersistAtom **- case of RT_SlidePersistAtom**
000067AE	000C	**I**: TextHeaderAtom **- case of RT_TextHeaderAtom**
000067BA	0016	**J**: TextBytesAtom **- case of RT_TextBytesAtom**
000067D0	000C	TextHeaderAtom **- case of RT_TextHeaderAtom**
000067DC	0012	TextSpecialInfoAtom **- case of RT_TextSpecialInfoAtom**
000067EE	001C	SlidePersistAtom **- case of RT_SlidePersistAtom**
0000680A	000C	**K**: TextHeaderAtom **- case of RT_TextHeaderAtom**
00006816	0015	**L**: TextBytesAtom **- case of RT_TextBytesAtom**
0000682B	001C	SlidePersistAtom **- case of RT_SlidePersistAtom**
00006847	000C	**M**: TextHeaderAtom **- case of RT_TextHeaderAtom**
00006853	0019	**N**: TextBytesAtom **- case of RT_TextBytesAtom**
0000686C	001C	SlidePersistAtom **- case of RT_SlidePersistAtom**
00006888	000C	**O**: TextHeaderAtom **- case of RT_TextHeaderAtom**
00006894	0018	**P**: TextBytesAtom **- case of RT_TextBytesAtom**

Figure 27: SlideListWithTextContainer child-record hierarchy

The character content of the outline text is derived from the TextBytesAtom records labeled B, D, F, H, J, L, N, and P in the previous table titled "SlideListWithTextContainer child-record hierarchy". The TextHeaderAtom records labeled A, C, E, G, I, K, M, and O that precede the TextBytesAtom records indicate the type of text content as specified by the **textType** field and the **TextTypeEnum.** These values can be used to organize the text into logical outline levels, where title placeholder shape text is a top-level outline item and body placeholder shape text is a subordinate outline item.

The detailed structures of records A through P in the previous table titled "SlideListWithTextContainer child-record hierarchy" are shown in the following tables. Non-printable character content in TextBytesAtom records has been replaced with "\t", "\n", "\v", or "\r" where applicable.

The first line of the outline text as shown in the figure titled "Outline text" in section 3.1 is derived from records A and B, which are shown expanded in the following tables.

Offset	Size	Structure	Value
000066BC	000C	**A**: TextHeaderAtom **- case of RT_TextHeaderAtom**	
000066BC	0008	RecordHeader **- rh**	
000066BC	4 bits	unsigned integer **- recVer**	0x0
000066BC	12 bits	unsigned integer **- recInstance**	0x000
000066BE	0002	RecordType **- recType**	0x0F9F

Offset	Size	Structure	Value
000066C0	0004	unsigned integer **- recLen**	0x00000004
000066C4	0004	TextTypeEnum **- textType**	0x00000006

Figure 28: Outline TextHeaderAtom record A

textType: 0x00000006 specifies that the following character content in the **textBytes** field of the following table represents the center title placeholder shape text.

Offset	Size	Structure	Value
000066C8	0010	**B**: TextBytesAtom **- case of RT_TextBytesAtom**	
000066C8	0008	RecordHeader **- rh**	
000066C8	4 bits	unsigned integer **- recVer**	0x0
000066C8	12 bits	unsigned integer **- recInstance**	0x000
000066CA	0002	RecordType **- recType**	0x0FA8
000066CC	0004	unsigned integer **- recLen**	0x00000008
000066D0	0008	array of bytes **- textBytes**	a sample

Figure 29: Outline TextBytesAtom record B

textBytes: "a sample" specifies the center title placeholder shape text.

The second line of the outline text as shown in the figure titled "Outline text" in section 3.1 is derived from records C and D, which are shown expanded in the following tables.

Offset	Size	Structure	Value
000066D8	000C	**C**: TextHeaderAtom **- case of RT_TextHeaderAtom**	
000066D8	0008	RecordHeader **- rh**	
000066D8	4 bits	unsigned integer **- recVer**	0x0
000066D8	12 bits	unsigned integer **- recInstance**	0x001
000066DA	0002	RecordType **- recType**	0x0F9F
000066DC	0004	unsigned integer **- recLen**	0x00000004
000066E0	0004	TextTypeEnum **- textType**	0x00000005

Figure 30: Outline TextHeaderAtom record C

textType: 0x00000005 specifies that the following character content in the **textBytes** field of the following table represents the center body placeholder shape text.

Offset	Size	Structure	Value
000066E4	0014	**D**: TextBytesAtom **- case of RT_TextBytesAtom**	
000066E4	0008	RecordHeader **- rh**	
000066E4	4 bits	unsigned integer **- recVer**	0x0

Offset	Size	Structure	Value
000066E4	12 bits	unsigned integer **- recInstance**	0x000
000066E6	0002	RecordType **- recType**	0x0FA8
000066E8	0004	unsigned integer **- recLen**	0x0000000C
000066EC	000C	array of bytes **- textBytes**	presentation

Figure 31: Outline TextBytesAtom record D

textBytes: "presentation" specifies the center body placeholder shape text.

The third line of the outline text as shown in the figure titled "Outline text" in section 3.1 is derived from records E and F, which are shown expanded in the following tables.

Offset	Size	Structure	Value
00006714	000C	**E**: TextHeaderAtom **- case of RT_TextHeaderAtom**	
00006714	0008	RecordHeader **- rh**	
00006714	4 bits	unsigned integer **- recVer**	0x0
00006714	12 bits	unsigned integer **- recInstance**	0x000
00006716	0002	RecordType **- recType**	0x0F9F
00006718	0004	unsigned integer **- recLen**	0x00000004
0000671C	0004	TextTypeEnum **- textType**	0x00000000

Figure 32: Outline TextHeaderAtom record E

textType: 0x00000000 specifies that the following character content in the **textBytes** field of the following table represents the title placeholder shape text.

Offset	Size	Structure	Value
00006720	0013	**F**: TextBytesAtom **- case of RT_TextBytesAtom**	
00006720	0008	RecordHeader **- rh**	
00006720	4 bits	unsigned integer **- recVer**	0x0
00006720	12 bits	unsigned integer **- recInstance**	0x000
00006722	0002	RecordType **- recType**	0x0FA8
00006724	0004	unsigned integer **- recLen**	0x0000000B
00006728	000B	array of bytes **- textBytes**	the weather

Figure 33: Outline TextBytesAtom record F

textBytes: "the weather" specifies the title placeholder shape text.

Lines four through six of the outline text as shown in the figure titled "Outline text" in section 3.1 are derived from records G and H, which are shown expanded in the following tables.

Offset	Size	Structure	Value
00006733	000C	**G**: TextHeaderAtom **- case of RT_TextHeaderAtom**	
00006733	0008	RecordHeader **- rh**	
00006733	4 bits	unsigned integer **- recVer**	0x0
00006733	12 bits	unsigned integer **- recInstance**	0x001
00006735	0002	RecordType **- recType**	0x0F9F
00006737	0004	unsigned integer **- recLen**	0x00000004
0000673B	0004	TextTypeEnum **- textType**	0x00000001

Figure 34: Outline TextHeaderAtom record G

textType: 0x00000001 specifies that the following character content in the **textBytes** field of the following table represents the body placeholder shape text.

Offset	Size	Structure	Value
0000673F	0031	**H**: TextBytesAtom **- case of RT_TextBytesAtom**	
0000673F	0008	RecordHeader **- rh**	
0000673F	4 bits	unsigned integer **- recVer**	0x0
0000673F	12 bits	unsigned integer **- recInstance**	0x000
00006741	0002	RecordType **- recType**	0x0FA8
00006743	0004	unsigned integer **- recLen**	0x00000029
00006747	0029	array of bytes **- textBytes**	a sunny day\rthe blue sky\rsome green grass

Figure 35: Outline TextBytesAtom record H

textBytes: "a sunny day\rthe blue sky\rsome green grass" specifies the body placeholder shape text. Each line break in the text, shown as "\r", is displayed as a separate outline item shown in the figure titled "Outline text" in section 3.1.

Line seven of the outline text as shown in the figure titled "Outline text" in section 3.1 is derived from records I and J, which are shown expanded in the following tables.

Offset	Size	Structure	Value
000067AE	000C	**I**: TextHeaderAtom **- case of RT_TextHeaderAtom**	
000067AE	0008	RecordHeader **- rh**	
000067AE	4 bits	unsigned integer **- recVer**	0x0
000067AE	12 bits	unsigned integer **- recInstance**	0x000
000067B0	0002	RecordType **- recType**	0x0F9F
000067B2	0004	unsigned integer **- recLen**	0x00000004
000067B6	0004	TextTypeEnum **- textType**	0x00000000

Figure 36: Outline TextHeaderAtom record I

textType: 0x00000000 specifies that the following character content in the **textBytes** field of the following table represents the title placeholder shape text.

Offset	Size	Structure	Value
000067BA	0016	**J**: TextBytesAtom **- case of RT_TextBytesAtom**	
000067BA	0008	RecordHeader **- rh**	
000067BA	4 bits	unsigned integer **- recVer**	0x0
000067BA	12 bits	unsigned integer **- recInstance**	0x000
000067BC	0002	RecordType **- recType**	0x0FA8
000067BE	0004	unsigned integer **- recLen**	0x0000000E
000067C2	000E	array of bytes **- textBytes**	a green square

Figure 37: Outline TextBytesAtom record J

textBytes: "a green square" specifies the title placeholder shape text.

Line eight of the outline text as shown in the figure titled "Outline text" in section 3.1 is derived from records K and L, which are shown expanded in the following tables.

Offset	Size	Structure	Value
0000680A	000C	**K**: TextHeaderAtom **- case of RT_TextHeaderAtom**	
0000680A	0008	RecordHeader **- rh**	
0000680A	4 bits	unsigned integer **- recVer**	0x0
0000680A	12 bits	unsigned integer **- recInstance**	0x000
0000680C	0002	RecordType **- recType**	0x0F9F
0000680E	0004	unsigned integer **- recLen**	0x00000004
00006812	0004	TextTypeEnum **- textType**	0x00000000

Figure 38: Outline TextHeaderAtom record K

textType: 0x00000000 specifies that the following character content in the **textBytes** field of the following table represents the title placeholder shape text.

Offset	Size	Structure	Value
00006816	0015	**L**: TextBytesAtom **- case of RT_TextBytesAtom**	
00006816	0008	RecordHeader **- rh**	
00006816	4 bits	unsigned integer **- recVer**	0x0
00006816	12 bits	unsigned integer **- recInstance**	0x000
00006818	0002	RecordType **- recType**	0x0FA8
0000681A	0004	unsigned integer **- recLen**	0x0000000D
0000681E	000D	array of bytes **- textBytes**	a hungry bear

Figure 39: Outline TextBytesAtom record L

textBytes: "a hungry bear" specifies the title placeholder shape text.

Line nine of the outline text as shown in the figure titled "Outline text" in section 3.1 is derived from records M and N, which are shown expanded in the following tables.

Offset	Size	Structure	Value
00006847	000C	**M**: TextHeaderAtom **- case of RT_TextHeaderAtom**	
00006847	0008	RecordHeader **- rh**	
00006847	4 bits	unsigned integer **- recVer**	0x0
00006847	12 bits	unsigned integer **- recInstance**	0x000
00006849	0002	RecordType **- recType**	0x0F9F
0000684B	0004	unsigned integer **- recLen**	0x00000004
0000684F	0004	TextTypeEnum **- textType**	0x00000000

Figure 40: Outline TextHeaderAtom record M

textType: 0x00000000 specifies that the following character content in the **textBytes** field of the following table represents the title placeholder shape text.

Offset	Size	Structure	Value
00006853	0019	**N**: TextBytesAtom **- case of RT_TextBytesAtom**	
00006853	0008	RecordHeader **- rh**	
00006853	4 bits	unsigned integer **- recVer**	0x0
00006853	12 bits	unsigned integer **- recInstance**	0x000
00006855	0002	RecordType **- recType**	0x0FA8
00006857	0004	unsigned integer **- recLen**	0x00000011
0000685B	0011	array of bytes **- textBytes**	a fruit pie chart

Figure 41: Outline TextBytesAtom record N

textBytes: "a fruit pie chart" specifies the title placeholder shape text.

The last line of the outline text as shown in the figure titled "Outline text" in section 3.1 is derived from records O and P, which are shown expanded in the following tables.

Offset	Size	Structure	Value
00006888	000C	**O**: TextHeaderAtom **- case of RT_TextHeaderAtom**	
00006888	0008	RecordHeader **- rh**	
00006888	4 bits	unsigned integer **- recVer**	0x0
00006888	12 bits	unsigned integer **- recInstance**	0x000
0000688A	0002	RecordType **- recType**	0x0F9F
0000688C	0004	unsigned integer **- recLen**	0x00000004

Offset	Size	Structure	Value
00006890	0004	TextTypeEnum **- textType**	0x00000000

Figure 42: Outline TextHeaderAtom record O

textType: 0x00000000 specifies that the following character content in the **textBytes** field of the following table represents the title placeholder shape text.

Offset	Size	Structure	Value
00006894	0018	**P**: TextBytesAtom **- case of RT_TextBytesAtom**	
00006894	0008	RecordHeader **- rh**	
00006894	4 bits	unsigned integer **- recVer**	0x0
00006894	12 bits	unsigned integer **- recInstance**	0x000
00006896	0002	RecordType **- recType**	0x0FA8
00006898	0004	unsigned integer **- recLen**	0x00000010
0000689C	0010	array of bytes **- textBytes**	shapes with text

Figure 43: Outline TextBytesAtom record P

textBytes: "shapes with text" specifies the title placeholder shape text.

For more information about how the outline text formatting as shown in the figure titled "Outline text" in section 3.1, for example bullet points, font face, and font size, is derived, see the Text Example section.

3.5 Slides Example

The following sections provide examples of a main master slide, a title master slide, a notes master slide, presentation slides, and notes slides.

3.5.1 Master Slides Example

This example explains how to locate the main master slide, the title master slide, and the notes master slide.

The main master slide and title master slide are found by means of the **MasterPersistAtom** records (section 2.4.14.2) contained within the **MasterListWithTextContainer** record (section 2.4.14.1). The child-record hierarchy of the **MasterListWithTextContainer** record A from the table titled "DocumentContainer child-record hierarchy" in section 3.4 is shown expanded in the following table.

Offset	Size	Structure
00006159	0040	**A**: MasterListWithTextContainer **- masterList**
00006159	0008	RecordHeader **- rh**
00006161	001C	**B**: MasterPersistAtom **- masterPersistAtom**
0000617D	001C	**C**: MasterPersistAtom **- masterPersistAtom**

Figure 44: MasterListWithTextContainer record A child-record hierarchy

The child-record hierarchy of the **MasterPersistAtom** record B from previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00006161	001C	**B**: MasterPersistAtom **- masterPersistAtom**	
00006161	0008	RecordHeader **- rh**	
00006169	0004	PersistIdRef **- persistIdRef**	0x00000002
0000616D	2 bits	unsigned integer **- reserved1**	0x0
0000616D	1 bit	bit **- fNonOutlineData**	0x0
0000616D	29 bits	unsigned integer **- reserved2**	0x00000000
00006171	0004	signed integer **- reserved3**	0x00000000
00006175	0004	MasterId **- masterId**	0x80000000
00006179	0004	unsigned integer **- reserved4**	0x00000000

Figure 45: MasterPersistAtom record B child-record hierarchy

persistIdRef: 0x00000002 specifies the value to look up in the persist object directory, shown in the last table in section 3.2, to find the persist object stream offset 0x00000CC5. This offset matches the offset for the **MainMasterContainer** (section 2.5.3) record B in the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3. This **MainMasterContainer** record represents the main master slide as shown in figure titled "Main master slide" in section 3.1.

The child-record hierarchy of the **MasterPersistAtom** record C from the previous table titled "MasterListWithTextContainer record A child-record hierarchy" is shown expanded in the following table.

Offset	Size	Structure	Value
0000617D	001C	**C**: MasterPersistAtom **- masterPersistAtom**	
0000617D	0008	RecordHeader **- rh**	
00006185	0004	PersistIdRef **- persistIdRef**	0x00000003
00006189	2 bits	unsigned integer **- reserved1**	0x0
00006189	1 bit	bit **- fNonOutlineData**	0x0
00006189	29 bits	unsigned integer **- reserved2**	0x00000000
0000618D	0004	signed integer **- reserved3**	0x00000000
00006191	0004	MasterId **- masterId**	0x80000001
00006195	0004	unsigned integer **- reserved4**	0x00000000

Figure 46: MasterPersistAtom record C child-record hierarchy

persistIdRef: 0x00000003 specifies the value to look up in the persist object directory, shown in the last table in section 3.2, to find the persist object stream offset 0x00001671. This offset matches the offset for the **SlideContainer** (section 2.5.1) the record C in the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3. This

SlideContainer record represents the title master slide as shown in figure titled "Title master slide" in section 3.1.

The notes master slide is found by means of the **DocumentAtom** record (section 2.4.2) contained within the **DocumentContainer** record (section 2.4.1). The **DocumentAtom** record E from the table titled "DocumentContainer child-record hierarchy" in section 3.4 is shown expanded in the following table.

Offset	Size	Structure	Value
00005C71	0030	**E**: DocumentAtom **- documentAtom**	
00005C71	0008	RecordHeader **- rh**	
00005C79	0008	PointStruct **- slideSize**	
00005C81	0008	PointStruct **- notesSize**	
00005C89	0008	RatioStruct **- serverZoom**	
00005C91	0004	PersistIdRef **- notesMasterPersistIdRef**	0x0000000B
00005C95	0004	PersistIdRef **- handoutMasterPersistIdRef**	0x00000000
00005C99	0002	unsigned integer **- firstSlideNumber**	0x0001
00005C9B	0002	SlideSizeEnum **- slideSizeType**	0x0000
00005C9D	0001	bool1 **- fSaveWithFonts**	0x00
00005C9E	0001	bool1 **- fOmitTitlePlace**	0x00
00005C9F	0001	bool1 **- fRightToLeft**	0x00
00005CA0	0001	bool1 **- fShowComments**	0x01

Figure 47: DocumentAtom record E child-record hierarchy

notesMasterPersistIdRef: 0x0000000B specifies the value to look up in the persist object directory, shown in the last table in section 3.2, to find the persist object stream offset 0x00001B98. This offset matches the offset for the **NotesContainer** (section 2.5.6) record D in section 3.3. This **NotesContainer** record (section 2.5.6) represents the notes master slide as shown in figure titled "Notes master slide" in section 3.1.

3.5.2 Presentation Slides Example

The presentation slides are found by means of the **SlidePersistAtom** records (section 2.4.14.5) contained within the **SlideListWithTextContainer** record (section 2.4.14.3). The child-record hierarchy of the **SlideListWithTextContainer** (section 2.4.14.3) record B from the table titled "DocumentContainer child-record hierarchy" in section 3.4 is shown expanded in the following table.

Offset	Size	Structure
00006698	0214	SlideListWithTextContainer **- slideList**
00006698	0008	RecordHeader **- rh**
000066A0	001C	**A**: SlidePersistAtom **- case of RT_SlidePersistAtom**
000066BC	000C	TextHeaderAtom **- case of RT_TextHeaderAtom**

Offset	Size	Structure
000066C8	0010	TextBytesAtom **- case of RT_TextBytesAtom**
000066D8	000C	TextHeaderAtom **- case of RT_TextHeaderAtom**
000066E4	0014	TextBytesAtom **- case of RT_TextBytesAtom**
000066F8	001C	**B**: SlidePersistAtom **- case of RT_SlidePersistAtom**
00006714	000C	TextHeaderAtom **- case of RT_TextHeaderAtom**
00006720	0013	TextBytesAtom **- case of RT_TextBytesAtom**
00006733	000C	TextHeaderAtom **- case of RT_TextHeaderAtom**
0000673F	0031	TextBytesAtom **- case of RT_TextBytesAtom**
00006770	0022	StyleTextPropAtom **- case of RT_StyleTextPropAtom**
00006792	001C	**C**: SlidePersistAtom **- case of RT_SlidePersistAtom**
000067AE	000C	TextHeaderAtom **- case of RT_TextHeaderAtom**
000067BA	0016	TextBytesAtom **- case of RT_TextBytesAtom**
000067D0	000C	TextHeaderAtom **- case of RT_TextHeaderAtom**
000067DC	0012	TextSpecialInfoAtom **- case of RT_TextSpecialInfoAtom**
000067EE	001C	**D**: SlidePersistAtom **- case of RT_SlidePersistAtom**
0000680A	000C	TextHeaderAtom **- case of RT_TextHeaderAtom**
00006816	0015	TextBytesAtom **- case of RT_TextBytesAtom**
0000682B	001C	**E**: SlidePersistAtom **- case of RT_SlidePersistAtom**
00006847	000C	TextHeaderAtom **- case of RT_TextHeaderAtom**
00006853	0019	TextBytesAtom **- case of RT_TextBytesAtom**
0000686C	001C	**F**: SlidePersistAtom **- case of RT_SlidePersistAtom**
00006888	000C	TextHeaderAtom **- case of RT_TextHeaderAtom**
00006894	0018	TextBytesAtom **- case of RT_TextBytesAtom**

Figure 48: SlideListWithTextContainer child-record hierarchy

The child-record hierarchy of the **SlidePersistAtom** record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
000066A0	001C	**A**: SlidePersistAtom **- case of RT_SlidePersistAtom**	
000066A0	0008	RecordHeader **- rh**	
000066A8	0004	PersistIdRef **- persistIdRef**	0x00000004
000066AC	1 bit	bit **- reserved1**	0x0
000066AC	1 bit	bit **- fShouldCollapse**	0x0

Offset	Size	Structure	Value
000066AC	1 bit	bit **- fNonOutlineData**	0x1
000066AC	29 bits	unsigned integer **- reserved2**	0x00000000
000066B0	0004	signed integer **- cTexts**	0x00000002
000066B4	0004	SlideId **- slideId**	0x00000100
000066B8	0004	unsigned integer **- reserved3**	0x00000000

Figure 49: SlidePersistAtom record A child-record hierarchy

persistIdRef: 0x00000004 specifies the value to look up in the persist object directory, shown in the last table in section 3.2, to find the persist object stream offset 0x000021C8. This offset matches the offset for the **SlideContainer** (section 2.5.1) record E in the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3. This **SlideContainer** record represents the first presentation slide as shown in figure titled "Presentation slide 1" in section 3.1.

The child-record hierarchy of **SlideContainer** record E from the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3 is shown expanded in the following table.

Offset	Size	Structure	Value
000021C8	022C	**E**: SlideContainer	
000021C8	0008	RecordHeader **- rh**	
000021D0	0020	SlideAtom **- slideAtom**	
000021D0	0008	RecordHeader **- rh**	
000021D8	0004	SlideLayoutType **- geom**	0x00000000
000021DC	0001	PlaceholderEnum **- pt**	0x0F
000021DD	0001	PlaceholderEnum **- pt**	0x10
000021DE	0001	PlaceholderEnum **- pt**	0x00
000021DF	0001	PlaceholderEnum **- pt**	0x00
000021E0	0001	PlaceholderEnum **- pt**	0x00
000021E1	0001	PlaceholderEnum **- pt**	0x00
000021E2	0001	PlaceholderEnum **- pt**	0x00
000021E3	0001	PlaceholderEnum **- pt**	0x00
000021E4	0004	MasterIdRef **- masterIdRef**	0x80000001
000021E8	0004	NotesIdRef **- notesIdRef**	0x00000105
000021EC	0002	SlideFlags **- slideFlags**	
000021EC	1 bit	bit **- fMasterObjects**	0x1
000021EC	1 bit	bit **- fMasterScheme**	0x1

Offset	Size	Structure	Value
000021EC	1 bit	bit **- fMasterBackground**	0x1
000021EC	13 bits	unsigned integer **- reserved**	0x0000
000021EE	0002	unsigned integer **- unused**	0x3014
000021F0	019C	DrawingContainer **- drawing**	
0000238C	0028	SlideSchemeColorSchemeAtom **- slideSchemeColorSchemeAtom**	
000023B4	0040	SlideProgTagsContainer **- slideProgTagsContainer**	

Figure 50: SlideContainer record E child-record hierarchy

slideAtom.masterIdRef: 0x80000001 specifies a reference to the **masterId** field in the table titled "MasterPersistAtom record C child-record hierarchy" in section 3.5.1, the **MasterPersistAtom** (section 2.4.14.2) for the title master slide. This specifies that this presentation slide follows the title master slide.

slideAtom.slideFlags.fMasterBackground: 0x0001 specifies that this presentation slide will inherit the background from the title master slide as shown in figure titled "Title master slide" in section 3.1.

The child-record hierarchy of the **SlidePersistAtom** record B from the table titled "SlideListWithTextContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
000066F8	001C	**B**: SlidePersistAtom **- case of RT_SlidePersistAtom**	
000066F8	0008	RecordHeader **- rh**	
00006700	0004	PersistIdRef **- persistIdRef**	0x00000005
00006704	1 bit	bit **- reserved1**	0x0
00006704	1 bit	bit **- fShouldCollapse**	0x0
00006704	1 bit	bit **- fNonOutlineData**	0x1
00006704	29 bits	unsigned integer **- reserved2**	0x00000000
00006708	0004	signed integer **- cTexts**	0x00000002
0000670C	0004	SlideId **- slideId**	0x00000101
00006710	0004	unsigned integer **- reserved3**	0x00000000

Figure 51: SlidePersistAtom record B child-record hierarchy

persistIdRef: 0x00000005 specifies the value to look up in the persist object directory, shown in the last table in section 3.2, to find the persist object stream offset 0x000023F4. This offset matches the offset for the **SlideContainer** (section 2.5.1) record F in the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3. This **SlideContainer** record represents the second presentation slide as shown in figure titled "Presentation slide 2" in section 3.1.

The child-record hierarchy of **SlideContainer** record F from the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3 is shown expanded in the following table.

Offset	Size	Structure	Value
000023F4	0A8D	**F**: SlideContainer	
000023F4	0008	RecordHeader **- rh**	
000023FC	0020	SlideAtom **- slideAtom**	
000023FC	0008	RecordHeader **- rh**	
00002404	0004	SlideLayoutType **- geom**	0x00000001
00002408	0001	PlaceholderEnum **- pt**	0x0D
00002409	0001	PlaceholderEnum **- pt**	0x0E
0000240A	0001	PlaceholderEnum **- pt**	0x00
0000240B	0001	PlaceholderEnum **- pt**	0x00
0000240C	0001	PlaceholderEnum **- pt**	0x00
0000240D	0001	PlaceholderEnum **- pt**	0x00
0000240E	0001	PlaceholderEnum **- pt**	0x00
0000240F	0001	PlaceholderEnum **- pt**	0x00
00002410	0004	unsigned integer **- masterIdRef**	0x80000000
00002414	0004	unsigned integer **- notesIdRef**	0x00000104
00002418	0002	SlideFlags **- slideFlags**	
00002418	1 bit	bit **- fMasterObjects**	0x1
00002418	1 bit	bit **- fMasterScheme**	0x1
00002418	1 bit	bit **- fMasterBackground**	0x1
00002418	13 bits	unsigned integer **- reserved**	0x0000
0000241A	0002	unsigned integer **- unused**	0x3014
0000241C	01C8	DrawingContainer **- drawing**	
000025E4	0028	SlideSchemeColorSchemeAtom **- slideSchemeColorSchemeAtom**	
0000260C	0875	**A**: SlideProgTagsContainer **- slideProgTagsContainer**	

Figure 52: SlideContainer record F child-record hierarchy

slideAtom.masterIdRef: 0x80000000 specifies a reference to the **masterId** field in the table titled "MasterPersistAtom record B child-record hierarchy" in section 3.5.1, the **MasterPersistAtom** for the main master slide. This specifies that this presentation slide follows the main master slide.

slideAtom.slideFlags.fMasterBackground: 0x0001 specifies this presentation slide will inherit the background from the main master slide as shown in figure titled "Main master slide" in section 3.1.

The child-record hierarchy of the **SlidePersistAtom** record C from the table titled "SlideListWithTextContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00006792	001C	**C**: SlidePersistAtom **- case of RT_SlidePersistAtom**	
00006792	0008	RecordHeader **- rh**	
0000679A	0004	PersistIdRef **- persistIdRef**	0x00000006
0000679E	1 bit	bit **- reserved1**	0x0
0000679E	1 bit	bit **- fShouldCollapse**	0x0
0000679E	1 bit	bit **- fNonOutlineData**	0x1
0000679E	29 bits	unsigned integer **- reserved2**	0x00000000
000067A2	0004	signed integer **- cTexts**	0x00000002
000067A6	0004	SlideId **- slideId**	0x00000102
000067AA	0004	unsigned integer **- reserved3**	0x00000000

Figure 53: SlidePersistAtom record C child-record hierarchy

persistIdRef: 0x00000006 specifies the value to look up in the persist object directory, shown in the last table in section 3.2, to find the persist object stream offset 0x00006964. This offset matches the offset for the **SlideContainer** (section 2.5.1) record T in the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3. This **SlideContainer** record represents the third presentation slide as shown in figure titled "Presentation slide 3" in section 3.1.

The child-record hierarchy of **SlideContainer** record T from the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3 is shown expanded in the following table.

Offset	Size	Structure	Value
00006964	1192	**T**: SlideContainer	
00006964	0008	RecordHeader **- rh**	
0000696C	0020	SlideAtom **- slideAtom**	
0000696C	0008	RecordHeader **- rh**	
00006974	0004	SlideLayoutType **- geom**	0x00000001
00006978	0001	PlaceholderEnum **- pt**	0x0D
00006979	0001	PlaceholderEnum **- pt**	0x0E
0000697A	0001	PlaceholderEnum **- pt**	0x00
0000697B	0001	PlaceholderEnum **- pt**	0x00
0000697C	0001	PlaceholderEnum **- pt**	0x00
0000697D	0001	PlaceholderEnum **- pt**	0x00

Offset	Size	Structure	Value
0000697E	0001	PlaceholderEnum **- pt**	0x00
0000697F	0001	PlaceholderEnum **- pt**	0x00
00006980	0004	MasterIdRef **- masterIdRef**	0x80000000
00006984	0004	NotesIdRef **- notesIdRef**	0x00000103
00006988	0002	SlideFlags **- slideFlags**	
00006988	1 bit	bit **- fMasterObjects**	0x1
00006988	1 bit	bit **- fMasterScheme**	0x1
00006988	1 bit	bit **- fMasterBackground**	0x0
00006988	13 bits	unsigned integer **- reserved**	0x0000
0000698A	0002	unsigned integer **- unused**	0x3014
0000698C	020E	DrawingContainer **- drawing**	
00006B9A	0028	SlideSchemeColorSchemeAtom **- slideSchemeColorSchemeAtom**	
00006BC2	0F34	SlideProgTagsContainer **- slideProgTagsContainer**	

Figure 54: SlideContainer record T child-record hierarchy

slideAtom.masterIdRef: 0x80000000 specifies a reference to the **masterId** field in the table titled "MasterPersistAtom record B child-record hierarchy" in section 3.5.1, the **MasterPersistAtom** for the main master slide. This specifies that this presentation slide follows the main master slide.

slideAtom.slideFlags.fMasterBackground: 0x0000 specifies this slide does not inherit the background from the main master slide and provides its own background as shown in figure titled "Presentation slide 3" in section 3.1.

The child-record hierarchy of the **SlidePersistAtom** record D from the table titled "SlideListWithTextContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
000067EE	001C	**D**: SlidePersistAtom **- case of RT_SlidePersistAtom**	
000067EE	0008	RecordHeader **- rh**	
000067F6	0004	PersistIdRef **- persistIdRef**	0x00000008
000067FA	1 bit	bit **- reserved1**	0x0
000067FA	1 bit	bit **- fShouldCollapse**	0x0
000067FA	1 bit	bit **- fNonOutlineData**	0x1
000067FA	29 bits	unsigned integer **- reserved2**	0x00000000
000067FE	0004	signed integer **- cTexts**	0x00000001
00006802	0004	SlideId **- slideId**	0x00000103

Offset	Size	Structure	Value
00006806	0004	unsigned integer **- reserved3**	0x00000000

Figure 55: SlidePersistAtom record D child-record hierarchy

persistIdRef: 0x00000008 specifies the value to look up in the persist object directory, shown in the last table in section 3.2, to find the persist object stream offset 0x000037B5. This offset matches the offset for the **SlideContainer** (section 2.5.1) record I in the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3. This **SlideContainer** record represents the fourth presentation slide as shown in figure titled "Presentation slide 4" in section 3.1.

The child-record hierarchy of **SlideContainer** record I from the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3 is shown expanded in the following table.

Offset	Size	Structure	Value
000037B5	06CE	**I**: SlideContainer	
000037B5	0008	RecordHeader **- rh**	
000037BD	0020	SlideAtom **- slideAtom**	
000037BD	0008	RecordHeader **- rh**	
000037C5	0004	SlideLayoutType **- geom**	0x00000001
000037C9	0001	PlaceholderEnum **- pt**	0x0D
000037CA	0001	PlaceholderEnum **- pt**	0x13
000037CB	0001	PlaceholderEnum **- pt**	0x00
000037CC	0001	PlaceholderEnum **- pt**	0x00
000037CD	0001	PlaceholderEnum **- pt**	0x00
000037CE	0001	PlaceholderEnum **- pt**	0x00
000037CF	0001	PlaceholderEnum **- pt**	0x00
000037D0	0001	PlaceholderEnum **- pt**	0x00
000037D1	0004	MasterIdRef **- masterIdRef**	0x80000000
000037D5	0004	NotesIdRef **- notesIdRef**	0x00000102
000037D9	0002	SlideFlags **- slideFlags**	
000037D9	1 bit	bit **- fMasterObjects**	0x1
000037D9	1 bit	bit **- fMasterScheme**	0x1
000037D9	1 bit	bit **- fMasterBackground**	0x1
000037D9	13 bits	unsigned integer **- reserved**	0x0000
000037DB	0002	unsigned integer **- unused**	0x3014
000037DD	01BA	DrawingContainer **- drawing**	

Offset	Size	Structure	Value
00003997	0028	SlideSchemeColorSchemeAtom **- slideSchemeColorSchemeAtom**	
000039BF	04C4	SlideProgTagsContainer **- slideProgTagsContainer**	

Figure 56: SlideContainer record I child-record hierarchy

slideAtom.masterIdRef: 0x80000000 specifies a reference to the **masterId** field in the table titled "MasterPersistAtom record B child-record hierarchy" in section 3.5.1, the **MasterPersistAtom** for the main master slide. This specifies that this presentation slide follows the main master slide.

slideAtom.slideFlags.fMasterBackground: 0x0001 specifies that this presentation slide will inherit the background from the main master slide as shown in figure titled "Main master slide" in section 3.1.

The child-record hierarchy of the **SlidePersistAtom** record E from the table titled "SlideListWithTextContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
0000682B	001C	**E**: SlidePersistAtom **- case of RT_SlidePersistAtom**	
0000682B	0008	RecordHeader **- rh**	
00006833	0004	PersistIdRef **- persistIdRef**	0x00000009
00006837	1 bit	bit **- reserved1**	0x0
00006837	1 bit	bit **- fShouldCollapse**	0x0
00006837	1 bit	bit **- fNonOutlineData**	0x1
00006837	29 bits	unsigned integer **- reserved2**	0x00000000
0000683B	0004	signed integer **- cTexts**	0x00000001
0000683F	0004	SlideId **- slideId**	0x00000104
00006843	0004	unsigned integer **- reserved3**	0x00000000

Figure 57: SlidePersistAtom record E child-record hierarchy

persistIdRef: 0x00000009 specifies the value to look up in the persist object directory, shown in the last table in section 3.2, to find the persist object stream offset 0x00003E83. This offset matches the offset for the **SlideContainer** (section 2.5.1) record J in the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3. This **SlideContainer** record represents the fifth presentation slide as shown in figure titled "Presentation slide 5" in section 3.1.

The child-record hierarchy of **SlideContainer** record J from the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3 is shown expanded in the following table.

Offset	Size	Structure	Value
00003E83	0248	**J**: SlideContainer	
00003E83	0008	RecordHeader **- rh**	

Offset	Size	Structure	Value
00003E8B	0020	SlideAtom **- slideAtom**	
00003E8B	0008	RecordHeader **- rh**	
00003E93	0004	SlideLayoutType **- geom**	0x00000001
00003E97	0001	PlaceholderEnum **- pt**	0x0D
00003E98	0001	PlaceholderEnum **- pt**	0x14
00003E99	0001	PlaceholderEnum **- pt**	0x00
00003E9A	0001	PlaceholderEnum **- pt**	0x00
00003E9B	0001	PlaceholderEnum **- pt**	0x00
00003E9C	0001	PlaceholderEnum **- pt**	0x00
00003E9D	0001	PlaceholderEnum **- pt**	0x00
00003E9E	0001	PlaceholderEnum **- pt**	0x00
00003E9F	0004	MasterIdRef **- masterIdRef**	0x80000000
00003EA3	0004	NotesIdRef **- notesIdRef**	0x00000101
00003EA7	0002	SlideFlags **- slideFlags**	
00003EA7	1 bit	bit **- fMasterObjects**	0x1
00003EA7	1 bit	bit **- fMasterScheme**	0x1
00003EA7	1 bit	bit **- fMasterBackground**	0x1
00003EA7	13 bits	unsigned integer **- reserved**	0x0000
00003EA9	0002	unsigned integer **- unused**	0x3014
00003EAB	01B8	**A**: DrawingContainer **- drawing**	
00004063	0028	SlideSchemeColorSchemeAtom **- slideSchemeColorSchemeAtom**	
0000408B	0040	SlideProgTagsContainer **- slideProgTagsContainer**	

Figure 58: SlideContainer record J child-record hierarchy

slideAtom.masterIdRef: 0x80000000 specifies a reference to the **masterId** field in the table titled "MasterPersistAtom record B child-record hierarchy" in section 3.5.1, the **MasterPersistAtom** for the main master slide. This specifies that this presentation slide follows the main master slide.

slideAtom.slideFlags.fMasterBackground: 0x0001 specifies that this presentation slide will inherit the background from the main master slide as shown in figure titled "Main master slide" in section 3.1.

The child-record hierarchy of the **SlidePersistAtom** record F from the table titled "SlideListWithTextContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
0000686C	001C	**F**: SlidePersistAtom **- case of RT_SlidePersistAtom**	
0000686C	0008	RecordHeader **- rh**	
00006874	0004	PersistIdRef **- persistIdRef**	0x00000013
00006878	1 bit	bit **- reserved1**	0x0
00006878	1 bit	bit **- fShouldCollapse**	0x0
00006878	1 bit	bit **- fNonOutlineData**	0x1
00006878	29 bits	unsigned integer **- reserved2**	0x00000000
0000687C	0004	signed integer **- cTexts**	0x00000001
00006880	0004	SlideId **- slideId**	0x00000105
00006884	0004	unsigned integer **- reserved3**	0x00000000

Figure 59: SlidePersistAtom record F child-record hierarchy

persistIdRef: 0x00000013 specifies the value to look up in the persist object directory, shown in the last table in section 3.2, to find the persist object stream offset 0x00007AF6. This offset matches the offset for the **SlideContainer** (section 2.5.1) record U in the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3. This represents the sixth and final presentation slide as shown in figure titled "Presentation slide 6" in section 3.1.

The child-record hierarchy of **SlideContainer** record U from the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3 is shown expanded in the following table.

Offset	Size	Structure	Value
00007AF6	04ED	**U**: SlideContainer	
00007AF6	0008	RecordHeader **- rh**	
00007AFE	0020	SlideAtom **- slideAtom**	
00007AFE	0008	RecordHeader **- rh**	
00007B06	0004	SlideLayoutType **- geom**	0x00000007
00007B0A	0001	PlaceholderEnum **- pt**	0x0D
00007B0B	0001	PlaceholderEnum **- pt**	0x00
00007B0C	0001	PlaceholderEnum **- pt**	0x00
00007B0D	0001	PlaceholderEnum **- pt**	0x00
00007B0E	0001	PlaceholderEnum **- pt**	0x00
00007B0F	0001	PlaceholderEnum **- pt**	0x00
00007B10	0001	PlaceholderEnum **- pt**	0x00
00007B11	0001	PlaceholderEnum **- pt**	0x00
00007B12	0004	MasterIdRef **- masterIdRef**	0x80000000

Offset	Size	Structure	Value
00007B16	0004	NotesIdRef **- notesIdRef**	0x00000106
00007B1A	0002	SlideFlags **- slideFlags**	
00007B1A	1 bit	bit **- fMasterObjects**	0x1
00007B1A	1 bit	bit **- fMasterScheme**	0x1
00007B1A	1 bit	bit **- fMasterBackground**	0x1
00007B1A	13 bits	unsigned integer **- reserved**	0x0000
00007B1C	0002	unsigned integer **- unused**	0x3014
00007B1E	003A	**B**: PerSlideHeadersFootersContainer **- perSlideHFContainer**	
00007B58	0423	**A**: DrawingContainer **- drawing**	
00007F7B	0028	SlideSchemeColorSchemeAtom **- slideSchemeColorSchemeAtom**	
00007FA3	0040	SlideProgTagsContainer **- slideProgTagsContainer**	

Figure 60: SlideContainer record U child-record hierarchy

slideAtom.masterIdRef: 0x80000000 specifies a reference to the **masterId** field in the table titled "MasterPersistAtom record B child-record hierarchy" in section 3.5.1, the **MasterPersistAtom** for the main master slide. This specifies that this presentation slide follows the main master slide.

slideAtom.slideFlags.fMasterBackground: 0x0001 specifies that this presentation slide will inherit the background from the main master slide as shown in figure titled "Main master slide" in section 3.1.

3.5.3 Notes Slides Example

The notes slides are found by means of the **NotesPersistAtom** records (section 2.4.14.7) contained within the **NotesListWithTextContainer** record (section 2.4.14.6). Unlike in the previous example, where the order of the **SlidePersistAtom** records (section 2.4.14.5) contained within the **SlideListWithTextContainer** record (section 2.4.14.3) determines the order of the presentation slides, the order of the **NotesPersistAtom** records is not meaningful. A notes slide is associated with its presentation slide by means of the **slideIdRef** field in the **NotesContainer** record (section 2.5.6).

The child-record hierarchy of the **NotesListWithTextContainer** record C from the table titled "DocumentContainer child-record hierarchy" in section 3.4 is shown expanded in the following table.

Offset	Size	Structure
000068AC	00B0	NotesListWithTextContainer **- notesList**
000068AC	0008	RecordHeader **- rh**
000068B4	001C	**A**: NotesPersistAtom **- notesPersistAtom**
000068D0	001C	**B**: NotesPersistAtom **- notesPersistAtom**
000068EC	001C	**C**: NotesPersistAtom **- notesPersistAtom**
00006908	001C	**D**: NotesPersistAtom **- notesPersistAtom**
00006924	001C	**E**: NotesPersistAtom **- notesPersistAtom**

Offset	Size	Structure
00006940	001C	**F**: NotesPersistAtom **- notesPersistAtom**

Figure 61: NotesListWithTextContainer child-record hierarchy

The child-record hierarchy of the **NotesPersistAtom** record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
000068B4	001C	**A**: NotesPersistAtom **- notesPersistAtom**	
000068B4	0008	RecordHeader **- rh**	
000068BC	0004	PersistIdRef **- persistIdRef**	0x0000000D
000068C0	2 bits	unsigned integer **- reserved1**	0x0
000068C0	1 bit	bit **- fNonOutlineData**	0x0
000068C0	29 bits	unsigned integer **- reserved2**	0x00000000
000068C4	0004	signed integer **- reserved3**	0x00000000
000068C8	0004	NotesId **- notesId**	0x00000101
000068CC	0004	unsigned integer **- reserved4**	0x00000000

Figure 62: NotesPersistAtom record A child-record hierarchy

persistIdRef: 0x0000000D specifies the value to look up in the persist object directory, shown in the last in section 3.2, to find the persist object stream offset 0x000040CB. This offset matches the offset for the **NotesContainer** (section 2.5.6) record K in the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3.

The child-record hierarchy of the **NotesContainer** (section 2.5.6) record K from the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3 is shown expanded in the following table.

Offset	Size	Structure	Value
000040CB	024E	**K**: NotesContainer	
000040CB	0008	RecordHeader **- rh**	
000040D3	0010	NotesAtom **- notesAtom**	
000040D3	0008	RecordHeader **- rh**	
000040DB	0004	SlideIdRef **- slideIdRef**	0x00000104
000040DF	0002	SlideFlags **- slideFlags**	
000040E3	01CE	DrawingContainer **- drawing**	
000042B1	0028	SlideSchemeColorSchemeAtom **- slideSchemeColorSchemeAtom**	
000042D9	0040	SlideProgTagsContainer **- slideProgTagsContainer**	

Figure 63: NotesContainer record K child-record hierarchy

notesAtom.slideIdRef: 0x00000104 specifies a reference to the **slideId** field in the table titled "SlidePersistAtom record E child-record hierarchy" in section 3.5.2, the **SlidePersistAtom** for the fifth presentation slide. This **NotesContainer** record (section 2.5.6) represents the fifth notes slide as shown in figure titled "Notes slide 5" in section 3.1.

The child-record hierarchy of the **NotesPersistAtom** record B from the table titled "NotesListWithTextContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
000068D0	001C	**B**: NotesPersistAtom **- notesPersistAtom**	
000068D0	0008	RecordHeader **- rh**	
000068D8	0004	PersistIdRef **- persistIdRef**	0x0000000E
000068DC	2 bits	unsigned integer **- reserved1**	0x0
000068DC	1 bit	bit **- fNonOutlineData**	0x0
000068DC	29 bits	unsigned integer **- reserved2**	0x00000000
000068E0	0004	signed integer **- reserved3**	0x00000000
000068E4	0004	NotesId **- notesId**	0x00000102
000068E8	0004	unsigned integer **- reserved4**	0x00000000

Figure 64: NotesPersistAtom record B child-record hierarchy

persistIdRef: 0x0000000E specifies the value to look up in the persist object directory, shown in the last table in section 3.2, to find the persist object stream offset 0x00004319. This offset matches the offset for the **NotesContainer** (section 2.5.6) record L in the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3.

The child-record hierarchy of the **NotesContainer** (section 2.5.6) record L from the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3 is shown expanded in the following table.

Offset	Size	Structure	Value
00004319	02BB	**L**: NotesContainer	
00004319	0008	RecordHeader **- rh**	
00004321	0010	NotesAtom **- notesAtom**	
00004321	0008	RecordHeader **- rh**	
00004329	0004	SlideIdRef **- slideIdRef**	0x00000103
0000432D	0002	SlideFlags **- slideFlags**	
00004331	023B	DrawingContainer **- drawing**	
0000456C	0028	SlideSchemeColorSchemeAtom **- slideSchemeColorSchemeAtom**	
00004594	0040	SlideProgTagsContainer **- slideProgTagsContainer**	

Figure 65: NotesContainer record L child-record hierarchy

notesAtom.slideIdRef: 0x00000103 specifies a reference to the **slideId** field in the table titled "SlidePersistAtom record D child-record hierarchy" in section 3.5.2, the **SlidePersistAtom** for the fourth presentation slide. This **NotesContainer** record (section 2.5.6) represents the fourth notes slide as shown in figure titled "Notes slide 4" in section 3.1.

The child-record hierarchy of the **NotesPersistAtom** record C from the table titled "NotesListWithTextContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
000068EC	001C	**C**: NotesPersistAtom **- notesPersistAtom**	
000068EC	0008	RecordHeader **- rh**	
000068F4	0004	PersistIdRef **- persistIdRef**	0x0000000F
000068F8	2 bits	unsigned integer **- reserved1**	0x0
000068F8	1 bit	bit **- fNonOutlineData**	0x0
000068F8	29 bits	unsigned integer **- reserved2**	0x00000000
000068FC	0004	signed integer **- reserved3**	0x00000000
00006900	0004	NotesId **- notesId**	0x00000103
00006904	0004	unsigned integer **- reserved4**	0x00000000

Figure 66: NotesPersistAtom record C child-record hierarchy

persistIdRef: 0x0000000F specifies the value to look up in the persist object directory, shown in the last table in section 3.2, to find the persist object stream offset 0x000045D4. This offset matches the offset for the **NotesContainer** (section 2.5.6) record M in the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3.

The child-record hierarchy of the **NotesContainer** (section 2.5.6) record M from the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3 is shown expanded in the following table.

Offset	Size	Structure	Value
000045D4	029F	**M**: NotesContainer	
000045D4	0008	RecordHeader **- rh**	
000045DC	0010	NotesAtom **- notesAtom**	
000045DC	0008	RecordHeader **- rh**	
000045E4	0004	SlideIdRef **- slideIdRef**	0x00000102
000045E8	0002	SlideFlags **- slideFlags**	
000045EC	021F	DrawingContainer **- drawing**	
0000480B	0028	SlideSchemeColorSchemeAtom **- slideSchemeColorSchemeAtom**	
00004833	0040	SlideProgTagsContainer **- slideProgTagsContainer**	

Figure 67: NotesContainer record M child-record hierarchy

notesAtom.slideIdRef: 0x00000102 specifies a reference to the **slideId** field in the table titled "SlidePersistAtom record C child-record hierarchy" in section 3.5.2, the **SlidePersistAtom** for the third presentation slide. This **NotesContainer** record (section 2.5.6) represents the third notes slide as shown in figure titled "Notes slide 3" in section 3.1.

The child-record hierarchy of the **NotesPersistAtom** record D from the table titled "NotesListWithTextContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00006908	001C	**D**: NotesPersistAtom **- notesPersistAtom**	
00006908	0008	RecordHeader **- rh**	
00006910	0004	PersistIdRef **- persistIdRef**	0x00000010
00006914	2 bits	unsigned integer **- reserved1**	0x0
00006914	1 bit	bit **- fNonOutlineData**	0x0
00006914	29 bits	unsigned integer **- reserved2**	0x00000000
00006918	0004	signed integer **- reserved3**	0x00000000
0000691C	0004	NotesId **- notesId**	0x00000104
00006920	0004	unsigned integer **- reserved4**	0x00000000

Figure 68: NotesPersistAtom record D child-record hierarchy

persistIdRef: 0x00000010 specifies the value to look up in the persist object directory, shown in the last table in section 3.2, to find the persist object stream offset 0x00004873. This offset matches the offset for the **NotesContainer** (section 2.5.6) record N in the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3.

The child-record hierarchy of the **NotesContainer** (section 2.5.6) record N from the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3 is shown expanded in the following table.

Offset	Size	Structure	Value
00004873	02AF	**N**: NotesContainer	
00004873	0008	RecordHeader **- rh**	
0000487B	0010	NotesAtom **- notesAtom**	
0000487B	0008	RecordHeader **- rh**	
00004883	0004	SlideIdRef **- slideIdRef**	0x00000101
00004887	0002	SlideFlags **- slideFlags**	
0000488B	022F	DrawingContainer **- drawing**	
00004ABA	0028	SlideSchemeColorSchemeAtom **- slideSchemeColorSchemeAtom**	
00004AE2	0040	SlideProgTagsContainer **- slideProgTagsContainer**	

Figure 69: NotesContainer record N child-record hierarchy

notesAtom.slideIdRef: 0x00000101 specifies a reference to the **slideId** field in the table titled "SlidePersistAtom record A child-record hierarchy" in section 3.5.2, the **SlidePersistAtom** for the second presentation slide. This **NotesContainer** record (section 2.5.6) represents the second notes slide as shown in figure titled "Notes slide 2" in section 3.1.

The child-record hierarchy of the **NotesPersistAtom** record E from the table titled "NotesListWithTextContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00006924	001C	**E**: NotesPersistAtom **- notesPersistAtom**	
00006924	0008	RecordHeader **- rh**	
0000692C	0004	PersistIdRef **- persistIdRef**	0x00000011
00006930	2 bits	unsigned integer **- reserved1**	0x0
00006930	1 bit	bit **- fNonOutlineData**	0x0
00006930	29 bits	unsigned integer **- reserved2**	0x00000000
00006934	0004	signed integer **- reserved3**	0x00000000
00006938	0004	NotesId **- notesId**	0x00000105
0000693C	0004	unsigned integer **- reserved4**	0x00000000

Figure 70: NotesPersistAtom record E child-record hierarchy

persistIdRef: 0x00000011 specifies the value to look up in the persist object directory, shown in the last table in section 3.2, to find the persist object stream offset 0x00004B22. This offset matches the offset for the **NotesContainer** (section 2.5.6) record O in the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3.

The child-record hierarchy of the **NotesContainer** (section 2.5.6) record O from the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3 is shown expanded in the following table.

Offset	Size	Structure	Value
00004B22	0247	**O**: NotesContainer	
00004B22	0008	RecordHeader **- rh**	
00004B2A	0010	NotesAtom **- notesAtom**	
00004B2A	0008	RecordHeader **- rh**	
00004B32	0004	SlideIdRef **- slideIdRef**	0x00000100
00004B36	0002	SlideFlags **- slideFlags**	
00004B3A	01C7	DrawingContainer **- drawing**	
00004D01	0028	SlideSchemeColorSchemeAtom **- slideSchemeColorSchemeAtom**	
00004D29	0040	SlideProgTagsContainer **- slideProgTagsContainer**	

Figure 71: NotesContainer record O child-record hierarchy

notesAtom.slideIdRef: 0x00000100 specifies a reference to the **slideId** field in the table titled "SlidePersistAtom record A child-record hierarchy" in section 3.5.2, the **SlidePersistAtom** for the first presentation slide. This **NotesContainer** record (section 2.5.6) represents the first notes slide as shown in figure titled "Notes slide 1" in section 3.1.

The child-record hierarchy of the **NotesPersistAtom** record F from the table titled "NotesListWithTextContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00006940	001C	**F**: NotesPersistAtom **- notesPersistAtom**	
00006940	0008	RecordHeader **- rh**	
00006948	0004	PersistIdRef **- persistIdRef**	0x00000014
0000694C	2 bits	unsigned integer **- reserved1**	0x0
0000694C	1 bit	bit **- fNonOutlineData**	0x0
0000694C	29 bits	unsigned integer **- reserved2**	0x00000000
00006950	0004	signed integer **- reserved3**	0x00000000
00006954	0004	NotesId **- notesId**	0x00000106
00006958	0004	unsigned integer **- reserved4**	0x00000000

Figure 72: NotesPersistAtom record F child-record hierarchy

persistIdRef: 0x00000014 specifies the value to look up in the persist object directory, shown in the last table in section 3.2, to find the persist object stream offset 0x00007FE3. This offset matches the offset for the **NotesContainer** (section 2.5.6) record V in the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3.

The child-record hierarchy of the **NotesContainer** (section 2.5.6) record V from the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3 is shown expanded in the following table.

Offset	Size	Structure	Value
00007FE3	0289	**V**: NotesContainer	
00007FE3	0008	RecordHeader **- rh**	
00007FEB	0010	NotesAtom **- notesAtom**	
00007FEB	0008	RecordHeader **- rh**	
00007FF3	0004	SlideIdRef **- slideIdRef**	0x00000105
00007FF7	0002	SlideFlags **- slideFlags**	
00007FFB	0209	DrawingContainer **- drawing**	
00008204	0028	SlideSchemeColorSchemeAtom **- slideSchemeColorSchemeAtom**	
0000822C	0040	SlideProgTagsContainer **- slideProgTagsContainer**	

Figure 73: NotesContainer record V child-record hierarchy

notesAtom.slideIdRef: 0x00000105 specifies a reference to the **slideId** field in the table titled "SlidePersistAtom record F child-record hierarchy" in section 3.5.2, the **SlidePersistAtom** for the sixth presentation slide. This **NotesContainer** record (section 2.5.6) represents the sixth notes slide as shown in figure titled "Notes slide 6" in section 3.1.

3.6 Programmable Tags Example

This file format allows for the storage of programmable tags at three levels—on the document itself, on each slide, and on each shape. While the primary purpose of programmable tags is to provide a mechanism for a third-party VBA **add-in** to persist opaque data in the file, they are also used as a way for a later version of PowerPoint to store new records in the file that will be safely ignored and preserved by an earlier version. The following sections show an example of this extension mechanism on the document and slide-levels.

3.6.1 Document Programmable Tags Example

Document-level programmable tags are found inside the **DocProgTagsContainer** record (section 2.4.23.1) contained within the **DocInfoListContainer** record (section 2.4.4). The **DocInfoListContainer** record D from the table titled "DocumentContainer child-record hierarchy" in section 3.4 is shown expanded in the following table.

Offset	Size	Structure
00006199	04D7	DocInfoListContainer **- docInfoList**
00006199	0008	RecordHeader **- rh**
000061A1	0024	NormalViewSetInfoContainer **- case of RT_NormalViewSetInfo9**
000061C5	00AF	SlideViewInfoInstance **- case of RT_SlideViewInfo**
00006274	0044	NotesTextViewInfoContainer **- case of RT_NotesTextViewInfo9**
000062B8	001C	VBAInfoContainer **- case of RT_VbaInfo**
000062D4	006F	SlideViewInfoInstance **- case of RT_SlideViewInfo**
00006343	0044	SorterViewInfoContainer **- case of RT_SorterViewInfo**
00006387	0044	OutlineViewInfoContainer **- case of RT_OutlineViewInfo**
000063CB	02A5	**A**: DocProgTagsContainer **- case of RT_ProgTags**

Figure 74: DocInfoListContainer child-record hierarchy

The **DocProgTagsContainer**record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
000063CB	02A5	DocProgTagsContainer **- case of RT_ProgTags**	
000063CB	0008	RecordHeader **- rh**	
000063D3	0038	**A**: DocProgBinaryTagContainer **- case of RT_ProgBinaryTag**	
000063D3	0008	RecordHeader **- rh**	
000063D3	4 bits	unsigned integer **- recVer**	0xF

Offset	Size	Structure	Value
000063D3	12 bits	unsigned integer **- recInstance**	0x000
000063D5	0002	RecordType **- recType**	0x138A
000063D7	0004	unsigned integer **- recLen**	0x00000030
000063DB	0030	**B**: PP10DocBinaryTagExtension **- case of ___PPT10**	
0000640B	0265	**C**: DocProgBinaryTagContainer **- case of RT_ProgBinaryTag**	
0000640B	0008	RecordHeader **- rh**	
0000640B	4 bits	unsigned integer **- recVer**	0xF
0000640B	12 bits	unsigned integer **- recInstance**	0x000
0000640D	0002	RecordType **- recType**	0x138A
0000640F	0004	unsigned integer **- recLen**	0x0000025D
00006413	025D	**D**: PP9DocBinaryTagExtension **- case of ___PPT9**	

Figure 75: DocProgTagsContainer child-record hierarchy

The **DocProgTagsContainer** record shown in the previous table has two programmable tags, represented by records labeled A and C, both of which store binary data as specified by the 0x138A value for their **rh.recType** fields.

The PP2.4.23.6DocBinaryTagExtension record B from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
000063DB	0030	**B**: PP10DocBinaryTagExtension **- case of ___PPT10**	
000063DB	0008	RecordHeader **- rh**	
000063DB	4 bits	unsigned integer **- recVer**	0x0
000063DB	12 bits	unsigned integer **- recInstance**	0x000
000063DD	0002	RecordType **- recType**	0x0FBA
000063DF	0004	unsigned integer **- recLen**	0x00000010
000063E3	0010	PrintableUnicodeString **- tagName**	___PPT10
000063F3	0008	RecordHeader **- rhData**	
000063F3	4 bits	unsigned integer **- recVer**	0x0
000063F3	12 bits	unsigned integer **- recInstance**	0x000
000063F5	0002	RecordType **- recType**	0x138B
000063F7	0004	unsigned integer **- recLen**	0x00000010
000063FB	0010	GridSpacing10Atom **- gridSpacingAtom**	

Figure 76: PP10DocBinaryTagExtension child-record hierarchy

Most records that have a **RecordHeader** structure (section 2.3.1) as the first field are either an atom record or a container record. However, the PP2.4.23.6DocBinaryTagExtension record is not a single atom record, but rather a pair of atom records and has two **RecordHeader** structures identified by the **rh** and **rhData** fields.

rh: The value of the **rh.recLen** field is 0x00000010 and specifies the size of all subsequent fields until the next **RecordHeader** structure.

tagName: "___PPT10" specifies that the binary tag data following the **rhData** field is as specified by the PP2.4.23.6DocBinaryTagExtension record.

rhData: The value of the **rhData.recVer** field is 0x0. Generally when the **recVer** field of a **RecordHeader** structure is not equal to 0xF, it indicates that the record that contains the **RecordHeader** structure is an atom record. However, in this context, **rhData** behaves more like a **RecordHeader** structure found as the first field of a container record, because following it are more atom records and container records, specifically a GridSpacing2.4.21.1Atom record. Accordingly, the value of the **rhData.recLen** field is 0x00000010 and specifies the size of all subsequent records to be included as part of the PP2.4.23.6DocBinaryTagExtension record.

The PP2.4.23.5DocBinaryTagExtension record D from the table titled "DocProgTagsContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00006413	025D	**D**: PP9DocBinaryTagExtension **- case of ___PPT9**	
00006413	0008	RecordHeader **- rh**	
00006413	4 bits	unsigned integer **- recVer**	0x0
00006413	12 bits	unsigned integer **- recInstance**	0x000
00006415	0002	RecordType **- recType**	0x0FBA
00006417	0004	unsigned integer **- recLen**	0x0000000E
0000641B	000E	PrintableUnicodeString **- tagName**	___PPT9
00006429	0008	RecordHeader **- rhData**	
00006429	4 bits	unsigned integer **- recVer**	0x0
00006429	12 bits	unsigned integer **- recInstance**	0x000
0000642B	0002	RecordType **- recType**	0x138B
0000642D	0004	unsigned integer **- recLen**	0x0000023F
00006431	01F1	BlipCollection9Container **- blipCollectionContainer**	
00006622	0020	ExHyperlink9Container **- exHyperlinkContainer**	
00006642	002E	OutlineTextProps9Container **- outlineTextPropsContainer**	

Figure 77: PP9DocBinaryTagExtension child-record hierarchy

Most records that have a **RecordHeader** structure as the first field are either an atom record or a container record. However, the PP2.4.23.5DocBinaryTagExtension record is not a single atom record, but rather a pair of atom records and has two **RecordHeader** structures identified by the **rh** and **rhData** fields.

rh: The value of the **rh.recLen** field is 0x0000000E and specifies the size of all subsequent fields until the next **RecordHeader** structure.

tagName: "___PPT9" specifies that the binary tag data following the **rhData** is as specified by the PP2.4.23.5DocBinaryTagExtension record.

rhData: The value of the **rhData.recVer** field is 0x0. Generally when the **recVer** field of a **RecordHeader** structure is not equal to 0xF, it indicates that the record that contains the **RecordHeader** structure is an atom record. However, in this context, **rhData** behaves more like a **RecordHeader** structure found as the first field of a container record, because following it are more atom records and container records, specifically a **BlipCollection9Container** record (section 2.9.72), a ExHyperlink2.10.21Container record, and a OutlineTextProps2.9.60Container record. Accordingly, the value of the **rhData.recLen** field is 0x0000023F and specifies the size of all subsequent records to be included as part of the PP2.4.23.5DocBinaryTagExtension record.

3.6.2 Slide Programmable Tags Example

Slide-level programmable tags are found inside the **SlideContainer** record (section 2.5.1). The SlideProgTagsContainer record A from the table titled "SlidePersistAtom record F child-record hierarchy" in section 3.5.2 is shown expanded in the following table.

Offset	Size	Structure	Value
0000260C	0875	SlideProgTagsContainer **- slideProgTagsContainer**	
0000260C	0008	RecordHeader **- rh**	
00002614	086D	**A**: SlideProgBinaryTagContainer **- case of RT_ProgBinaryTag**	
00002614	0008	RecordHeader **- rh**	
00002614	4 bits	unsigned integer **- recVer**	0xF
00002614	12 bits	unsigned integer **- recInstance**	0x000
00002616	0002	RecordType **- recType**	0x138A
00002618	0004	unsigned integer **- recLen**	0x00000865
0000261C	0865	**B**: PP10SlideBinaryTagExtension **- case of ___PPT10**	
0000261C	0008	**C**: RecordHeader **- rh**	
0000261C	4 bits	unsigned integer **- recVer**	0x0
0000261C	12 bits	unsigned integer **- recInstance**	0x000
0000261E	0002	RecordType **- recType**	0x0FBA
00002620	0004	unsigned integer **- recLen**	0x00000010
00002624	0010	PrintableUnicodeString **- tagName**	___PPT10
00002634	0008	RecordHeader **- rhData**	
00002634	4 bits	unsigned integer **- recVer**	0x0
00002634	12 bits	unsigned integer **- recInstance**	0x000
00002636	0002	RecordType **- recType**	0x138B
00002638	0004	unsigned integer **- recLen**	0x00000845

Offset	Size	Structure	Value
0000263C	0010	SlideTime10Atom **- slideTimeAtom**	
0000264C	000C	HashCode10Atom **- hashCodeAtom**	
00002658	07E9	**D**: ExtTimeNodeContainer **- extTimeNodeContainer**	
00002E41	0040	BuildListContainer **- buildListContainer**	

Figure 78: SlideProgTagsContainer child-record hierarchy

The SlideProgTagsContainer record shown in the previous table has one programmable tag, represented by the record A, which stores binary data as specified by the 0x138A value for its **rh.recType** fields.

The contents of the programmable tag are shown by record B. Most records that have a **RecordHeader** structure (section 2.3.1) as the first field are either an atom record or a container record. However, the PP2.5.24SlideBinaryTagExtension record B is not a single atom record, but rather a pair of atom records and has two **RecordHeader** structures identified by the **rh** and **rhData** fields.

case of RT_ProgBinaryTag.case of ___PPT10.rh: The value of the **rh.recLen** field of the **RecordHeader** structure C is 0x00000010 and specifies the size of all subsequent fields until the next **RecordHeader** structure.

case of RT_ProgBinaryTag.case of ___PPT10.tagName: "___PPT10" specifies that the binary tag data following the **rhData** field is as specified by the PP2.5.24SlideBinaryTagExtension record.

case of RT_ProgBinaryTag.case of ___PPT10.rhData: The value of the **rhData.recVer** field is 0x0. Generally when the **recVer** field of a **RecordHeader** structure is not equal to 0xF, it indicates that the record that contains the **RecordHeader** structure is an atom record. However, in this context, **rhData** behaves more like a **RecordHeader** structure found as the first field of a container record, because following it are more atom records and container records, specifically a SlideTime2.5.31Atom record, a HashCode2.8.3Atom record, an **ExtTimeNodeContainer** record (section 2.8.15), and a BuildListContainer record. Accordingly, the value of the **rhData.recLen** field is 0x00000845 and specifies the size of all subsequent records to be included as part of the PP2.5.24SlideBinaryTagExtension record.

3.7 Animation Example

The following sections provide examples of text animation and shape animation. The animation in each example is synchronized with a timing tree that is a six-level tree of time nodes. The first level is the root of the tree, which is stored inside the PP2.5.24SlideBinaryTagExtension record.

3.7.1 Text Animation Example

This example explains the timing tree of the text animation used on presentation slide 2 as shown in figure titled "Presentation slide 2" in section 3.1. The animation displays one bullet item of paragraph every time the mouse is clicked. In total, three paragraphs that are not displayed at first are displayed, using animation.

The timing tree of the text animation is found inside the PP2.5.24SlideBinaryTagExtension record. The **ExtTimeNodeContainer** (section 2.8.15) record D from the table titled "SlideProgTagsContainer child-record hierarchy" in section 3.6.2 is shown expanded in the following table.

Offset	Size	Structure	Value
00002658	07E9	ExtTimeNodeContainer **- extTimeNodeContainer**	

Offset	Size	Structure	Value
00002658	0008	RecordHeader **- rh**	
00002660	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
00002688	0015	TimePropertyList4TimeNodeContainer **- timePropertyList**	
00002688	0008	RecordHeader **- rh**	
00002690	000D	TimeEffectNodeType **- case of TL_TPID_EffectNodeType**	
00002690	0008	RecordHeader **- rh**	
00002698	0001	TimeVariantTypeEnum **- type**	0x01
00002699	0004	signed integer **- effectNodeType**	0x00000009
0000269D	07A4	**B**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	

Figure 79: First-level ExtTimeNodeContainer child-record hierarchy

timePropertyList.case of TL_TPID_EffectNodeType.effectNodeType: 0x00000009 specifies that this time node is the root of the timing tree.

The child-record hierarchy of the TimeNodeAtom record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00002660	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
00002660	0008	RecordHeader **- rh**	
00002668	0004	unsigned integer **- reserved1**	0x00000000
0000266C	0004	unsigned integer **- restart**	0x00000003
00002670	0004	TimeNodeTypeEnum **- type**	0x00000000
00002674	0004	unsigned integer **- fill**	0x00000000
00002678	0004	signed integer **- reserved2**	0x00000000
0000267C	0001	unsigned integer **- reserved3**	0x00
00002680	0004	signed integer **- duration**	0xFFFFFFFF
00002684	1 bit	bit **- fFillProperty**	0x0
00002684	1 bit	bit **- fRestartProperty**	0x1
00002684	1 bit	bit **- reserved4**	0x0
00002684	1 bit	bit **- fGroupingTypeProperty**	0x0
00002684	1 bit	bit **- fDurationProperty**	0x1
00002684	27 bits	unsigned integer **- reserved5**	0x0000000

Figure 80: TimeNodeAtom record A child-record hierarchy

restart: 0x00000003 specifies that this time node will never restart.

type: 0x00000000 specifies that this time node is a parallel time node, which allows all of its child nodes to start at the same time.

fill: 0x00000000 specifies that the animated properties are reset to their original values after the time node becomes inactive.

duration: 0xFFFFFFFF specifies that the duration of the time node is infinite, and that its actual duration is determined by the durations of its child nodes.

The child-record hierarchy of the **ExtTimeNodeContainer** record B from the first table titled "First-level ExtTimeNodeContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
0000269D	07A4	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
0000269D	0008	RecordHeader **- rh**	
000026A5	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
000026CD	0015	TimePropertyList4TimeNodeContainer **- timePropertyList**	
000026CD	0008	RecordHeader **- rh**	
000026D5	000D	TimeEffectNodeType **- case of TL_TPID_EffectNodeType**	
000026D5	0008	RecordHeader **- rh**	
000026DD	0001	TimeVariantTypeEnum **- type**	0x01
000026DE	0004	signed integer **- effectNodeType**	0x00000004
000026E2	001C	**B**: TimeSequenceDataAtom **- timeSequenceDataAtom**	
000026FE	0034	**C**: TimeConditionContainer **- timeCondition**	
00002732	0034	**D**: TimeConditionContainer **- timeCondition**	
00002766	0249	**E**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
000029AF	0249	**F**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
00002BF8	0249	**G**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	

Figure 81: Second-level ExtTimeNodeContainer child-record hierarchy

The **TimeConditionContainer** (section 2.8.75) record C specifies the time condition of starting the next child time node, while the **TimeConditionContainer** record D specifies the time condition of starting the previous child time node.

The **ExtTimeNodeContainer** records E, F, and G are three child nodes of this time node. Each of them specifies the animation of one paragraph. Because these records are very similar, only the record E will be explained in this example.

timePropertyList.case of TL_TPID_EffectNodeType.effectNodeType: 0x00000004 specifies that this time node is the main sequential time node.

The child-record hierarchy of the TimeNodeAtom record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
000026A5	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
000026A5	0008	RecordHeader **- rh**	
000026AD	0004	unsigned integer **- reserved1**	0x00000000
000026B1	0004	unsigned integer **- restart**	0x00000000
000026B5	0004	TimeNodeTypeEnum **- type**	0x00000001
000026B9	0004	unsigned integer **- fill**	0x00000000
000026BD	0004	signed integer **- reserved2**	0x00000000
000026C1	0001	unsigned integer **- reserved3**	0x00
000026C5	0004	signed integer **- duration**	0xFFFFFFFF
000026C9	1 bit	bit **- fFillProperty**	0x0
000026C9	1 bit	bit **- fRestartProperty**	0x0
000026C9	1 bit	bit **- reserved4**	0x0
000026C9	1 bit	bit **- fGroupingTypeProperty**	0x1
000026C9	1 bit	bit **- fDurationProperty**	0x1
000026C9	27 bits	unsigned integer **- reserved5**	0x0000000

Figure 82: TimeNodeAtom record A child-record hierarchy in the second-level ExtTimeNodeContainer

type: 0x00000001 specifies that this time node is a sequential time node.

The child-record hierarchy of the TimeSequenceDataAtom record B from the table titled "Second-level ExtTimeNodeContainer child-record hierarchy" in section 3.7.2 is shown expanded in the following table.

Offset	Size	Structure	Value
000026E2	001C	**B**: TimeSequenceDataAtom **- timeSequenceDataAtom**	
000026E2	0008	RecordHeader **- rh**	
000026EA	0004	unsigned integer **- concurrency**	0x00000001
000026EE	0004	unsigned integer **- nextAction**	0x00000001
000026F2	0004	unsigned integer **- previousAction**	0x00000000
000026F6	0004	unsigned integer **- reserved1**	0x00000000
000026FA	1 bit	bit **- fConcurrencyPropertyUsed**	0x1
000026FA	1 bit	bit **- fNextActionPropertyUsed**	0x1
000026FA	1 bit	bit **- fPreviousActionPropertyUsed**	0x0
000026FA	29 bits	unsigned integer **- reserved2**	0x00000000

Figure 83: TimeSequenceDataAtom record B child-record hierarchy

concurrency: 0x00000001 specifies that the next child time node can be activated after the current child is activated and that the time condition in the **TimeConditionContainer** record C from the table titled "Second-level ExtTimeNodeContainer child-record hierarchy" in section 3.7.2 are met.

nextAction: 0x00000001 specifies that the current child time node traverses forward to its natural end time before traversing to the next child.

previousAction: 0x00000000 specifies that no actions are performed before traversing back to the previous child time node.

The child-record hierarchy of the **TimeConditionContainer** record C from the table titled "Second-level ExtTimeNodeContainer child-record hierarchy" in section 3.7.2 is shown expanded in the following table.

Offset	Size	Structure	Value
000026FE	0034	**C**: TimeConditionContainer **- timeCondition**	
000026FE	0008	RecordHeader **- rh**	
00002706	0018	TimeConditionAtom **- conditionAtom**	
00002706	0008	RecordHeader **- rh**	
0000270E	0004	TriggerObjectEnum **- triggerObject**	0x00000001
00002712	0004	unsigned integer **- triggerEvent**	0x00000009
00002716	0004	unsigned integer **- id**	0x00000000
0000271A	0004	signed integer **- delay**	0x00000000
0000271E	0014	ClientVisualElementContainer **- visualElement**	
0000271E	0008	RecordHeader **- rh**	
00002726	000C	VisualElementAtom **- visualElementAtom**	
00002726	000C	VisualPageAtom **- case of RT_VisualPageAtom**	
00002726	0008	RecordHeader **- rh**	
0000272E	0004	TimeVisualElementEnum **- type**	0x00000001

Figure 84: TimeConditionContainer record C child-record hierarchy in the second-level ExtTimeNodeContainer

The **TimeConditionContainer** specifies the time condition that triggers the next child time node.

conditionAtom.triggerObject: 0x00000001 specifies that the target that participates in the evaluation of time condition is an object as specified in the **timeCondition.visualElement** field.

conditionAtom.triggerEvent: 0x00000009 specifies that if the OnNext event occurs on the target, the time condition is true. The OnNext event can be triggered by a mouse click on the presentation slide.

conditionAtom.id: 0x00000000 specifies that the target is the presentation slide.

conditionAtom.delay: 0x00000000 specifies that there is no offset time applied to when the time condition becomes true.

visualElement.visualElementAtom.case of RT_VisualPageAtom.type: 0x00000001 specifies that the target of the time condition is a presentation slide.

The child-record hierarchy of the **TimeConditionContainer** record D from the table titled "Second-level ExtTimeNodeContainer child-record hierarchy" in section 3.7.2 is shown expanded in the following table.

Offset	Size	Structure	Value
00002732	0034	**D**: TimeConditionContainer **- timeCondition**	
00002732	0008	RecordHeader **- rh**	
0000273A	0018	TimeConditionAtom **- conditionAtom**	
0000273A	0008	RecordHeader **- rh**	
00002742	0004	TriggerObjectEnum **- triggerObject**	0x00000001
00002746	0004	unsigned integer **- triggerEvent**	0x0000000A
0000274A	0004	unsigned integer **- id**	0x00000000
0000274E	0004	signed integer **- delay**	0x00000000
00002752	0014	ClientVisualElementContainer **- visualElement**	
00002752	0008	RecordHeader **- rh**	
0000275A	000C	VisualElementAtom **- visualElementAtom**	
0000275A	000C	VisualPageAtom **- case of RT_VisualPageAtom**	
0000275A	0008	RecordHeader **- rh**	
00002762	0004	TimeVisualElementEnum **- type**	0x00000001

Figure 85: TimeConditionContainer record D child-record hierarchy in the second-level ExtTimeNodeContainer

The **TimeConditionContainer** specifies the time condition that triggers the previous child time node. The only difference between the **TimeConditionContainer** record C from the table titled "Second-level ExtTimeNodeContainer child-record hierarchy" in section 3.7.2 and the **TimeConditionContainer** record D from the table titled "Second-level ExtTimeNodeContainer child-record hierarchy" in section 3.7.2 is the **timeCondition.conditionAtom.triggerEvent** field.

conditionAtom.triggerEvent: 0x0000000A specifies that if the OnPrev event occurs on the target, the time condition is true.

The child-record hierarchy of the **ExtTimeNodeContainer** record E from the table titled "Second-level ExtTimeNodeContainer child-record hierarchy" in section 3.7.2 is shown expanded in the following table.

Offset	Size	Structure
00002766	0249	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**
00002766	0008	RecordHeader **- rh**
0000276E	0028	**A**: TimeNodeAtom **- timeNodeAtom**
00002796	0008	**B**: TimePropertyList4TimeNodeContainer **- timePropertyList**
00002796	0008	RecordHeader **- rh**

Offset	Size	Structure
0000279E	0020	**C**: TimeConditionContainer **- timeCondition**
000027BE	01F1	**D**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**

Figure 86: Third-level ExtTimeNodeContainer child-record hierarchy

The **TimePropertyList4TimeNodeContainer** (section 2.8.18) record B is an empty list.

The child-record hierarchy of the TimeNodeAtom record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
0000276E	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
0000276E	0008	RecordHeader **- rh**	
00002776	0004	unsigned integer **- reserved1**	0x00000000
0000277A	0004	unsigned integer **- restart**	0x00000000
0000277E	0004	TimeNodeTypeEnum **- type**	0x00000000
00002782	0004	unsigned integer **- fill**	0x00000003
00002786	0004	signed integer **- reserved2**	0x00000000
0000278A	0001	unsigned integer **- reserved3**	0x00
0000278E	0004	signed integer **- duration**	0x00000000
00002792	1 bit	bit **- fFillProperty**	0x1
00002792	1 bit	bit **- fRestartProperty**	0x0
00002792	1 bit	bit **- reserved4**	0x0
00002792	1 bit	bit **- fGroupingTypeProperty**	0x0
00002792	1 bit	bit **- fDurationProperty**	0x0
00002792	27 bits	unsigned integer **- reserved5**	0x0000000

Figure 87: TimeNodeAtom record A child-record hierarchy in the third-level ExtTimeNodeContainer

fill: 0x00000003 specifies that the properties remain at their ending values while the parent time node at the second level is still running or holding.

duration: 0x00000000 specifies that the duration of the time node is not specified, and that its actual duration is determined by the latest end time of any of its child nodes.

The child-record hierarchy of the **TimeConditionContainer** record C from the first table titled "Third-level ExtTimeNodeContainer child-record hierarchy" in section 3.7.2 is shown expanded in the following table.

Offset	Size	Structure	Value
0000279E	0020	**C**: TimeConditionContainer **- timeCondition**	
0000279E	0008	RecordHeader **- rh**	

Offset	Size	Structure	Value
000027A6	0018	TimeConditionAtom **- conditionAtom**	
000027A6	0008	RecordHeader **- rh**	
000027AE	0004	TriggerObjectEnum **- triggerObject**	0x00000000
000027B2	0004	unsigned integer **- triggerEvent**	0x00000000
000027B6	0004	unsigned integer **- id**	0x00000000
000027BA	0004	signed integer **- delay**	0xFFFFFFFF

Figure 88: TimeConditionContainer record C child-record hierarchy in the third-level ExtTimeNodeContainer

The value of the **conditionAtom.delay** field is 0xFFFFFFFF and specifies an infinite value, which implies that no delay is defined. The value of the **conditionAtom.triggerObject** field and the value of the **conditionAtom.triggerEvent** field are both 0x00000000, which specifies no condition for this time node, which implies that the evaluation of the time condition always returns true.

The child-record hierarchy of the **ExtTimeNodeContainer** record D from the first table titled "Third-level ExtTimeNodeContainer child-record hierarchy" in section 3.7.2 is shown expanded in the following table.

Offset	Size	Structure	Value
000027BE	01F1	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
000027BE	0008	RecordHeader **- rh**	
000027C6	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
000027EE	0008	TimePropertyList4TimeNodeContainer **- timePropertyList**	
000027EE	0008	RecordHeader **- rh**	
000027F6	0020	**B**: TimeConditionContainer **- timeCondition**	
000027F6	0008	RecordHeader **- rh**	
000027FE	0018	TimeConditionAtom **- conditionAtom**	
000027FE	0008	RecordHeader **- rh**	
00002806	0004	TriggerObjectEnum **- triggerObject**	0x00000000
0000280A	0004	unsigned integer **- triggerEvent**	0x00000000
0000280E	0004	unsigned integer **- id**	0x00000000
00002812	0004	signed integer **- delay**	0x00000000
00002816	0199	**C**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	

Figure 89: Fourth-level ExtTimeNodeContainer child-record hierarchy

The values of the **conditionAtom.triggerObject** field and the **conditionAtom.triggerEvent** field in the **TimeConditionContainer** record B are both 0x00000000, which specifies no condition for this time node and implies that the evaluation of the time condition always returns true.

The child-record hierarchy of the TimeNodeAtom record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
000027C6	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
000027C6	0008	RecordHeader **- rh**	
000027CE	0004	unsigned integer **- reserved1**	0x00000000
000027D2	0004	unsigned integer **- restart**	0x00000000
000027D6	0004	TimeNodeTypeEnum **- type**	0x00000000
000027DA	0004	unsigned integer **- fill**	0x00000003
000027DE	0004	signed integer **- reserved2**	0x00000000
000027E2	0001	unsigned integer **- reserved3**	0x00
000027E6	0004	signed integer **- duration**	0x00000000
000027EA	1 bit	bit **- fFillProperty**	0x1
000027EA	1 bit	bit **- fRestartProperty**	0x0
000027EA	1 bit	bit **- reserved4**	0x0
000027EA	1 bit	bit **- fGroupingTypeProperty**	0x0
000027EA	1 bit	bit **- fDurationProperty**	0x0
000027EA	27 bits	unsigned integer **- reserved5**	0x0000000

Figure 90: TimeNodeAtom record A child-record hierarchy in the fourth-level ExtTimeNodeContainer

fill: 0x00000003 specifies that the properties remain at their ending values while the parent time node at the third level is still running or holding.

The child-record hierarchy of the **ExtTimeNodeContainer** record C from the table titled "Fourth-level ExtTimeNodeContainer child-record hierarchy" in section 3.7.2 is shown expanded in the following table.

Offset	Size	Structure	Value
00002816	0199	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
00002816	0008	RecordHeader **- rh**	
0000281E	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
00002846	0049	**B**: TimePropertyList4TimeNodeContainer **- timePropertyList**	
0000288F	0020	**C**: TimeConditionContainer **- timeCondition**	
0000288F	0008	RecordHeader **- rh**	
00002897	0018	TimeConditionAtom **- conditionAtom**	
00002897	0008	RecordHeader **- rh**	

Offset	Size	Structure	Value
0000289F	0004	TriggerObjectEnum **- triggerObject**	0x00000000
000028A3	0004	unsigned integer **- triggerEvent**	0x00000000
000028A7	0004	unsigned integer **- id**	0x00000000
000028AB	0004	signed integer **- delay**	0x00000000
000028AF	0100	**D**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	

Figure 91: Fifth-level ExtTimeNodeContainer child-record hierarchy

This time node specifies the appear animation effect that makes the paragraph display during a slide show.

The values of the **conditionAtom.triggerObject** field and the **conditionAtom.triggerEvent** field in the **TimeConditionContainer** record C are both 0x00000000, which specifies no condition for this time node and implies that the evaluation of the time condition always returns true.

The child-record hierarchy of the TimeNodeAtom record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
0000281E	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
0000281E	0008	RecordHeader **- rh**	
00002826	0004	unsigned integer **- reserved1**	0x00000000
0000282A	0004	unsigned integer **- restart**	0x00000000
0000282E	0004	TimeNodeTypeEnum **- type**	0x00000000
00002832	0004	unsigned integer **- fill**	0x00000003
00002836	0004	signed integer **- reserved2**	0x00000000
0000283A	0001	unsigned integer **- reserved3**	0x00
0000283E	0004	signed integer **- duration**	0x00000000
00002842	1 bit	bit **- fFillProperty**	0x1
00002842	1 bit	bit **- fRestartProperty**	0x0
00002842	1 bit	bit **- reserved4**	0x0
00002842	1 bit	bit **- fGroupingTypeProperty**	0x0
00002842	1 bit	bit **- fDurationProperty**	0x0
00002842	27 bits	unsigned integer **- reserved5**	0x0000000

Figure 92: TimeNodeAtom record A child-record hierarchy in the fifth-level ExtTimeNodeContainer

fill: 0x00000003 specifies that the properties remain at their ending values while the parent time node at the fourth level is still running or holding.

The child-record hierarchy of the **TimePropertyList4TimeNodeContainer** record B from the table titled "Fifth-level ExtTimeNodeContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00002846	0049	**B**: TimePropertyList4TimeNodeContainer **- timePropertyList**	
00002846	0008	RecordHeader **- rh**	
0000284E	000D	TimeEffectNodeType **- case of TL_TPID_EffectNodeType**	
0000284E	0008	RecordHeader **- rh**	
00002856	0001	TimeVariantTypeEnum **- type**	0x01
00002857	0004	signed integer **- effectNodeType**	0x00000001
0000285B	000D	TimeEffectID **- case of TL_TPID_EffectID**	
0000285B	0008	RecordHeader **- rh**	
00002863	0001	TimeVariantTypeEnum **- type**	0x01
00002864	0004	signed integer **- effectID**	0x00000001
00002868	000D	TimeVariantInt **- case of TL_TPID_EffectDir**	
00002868	0008	RecordHeader **- rh**	
00002870	0001	TimeVariantTypeEnum **- type**	0x01
00002871	0004	signed integer **- intValue**	0x00000000
00002875	000D	TimeEffectType **- case of TL_TPID_EffectType**	
00002875	0008	RecordHeader **- rh**	
0000287D	0001	TimeVariantTypeEnum **- type**	0x01
0000287E	0004	signed integer **- effectType**	0x00000001
00002882	000D	TimeGroupID **- case of TL_TPID_GroupID**	
00002882	0008	RecordHeader **- rh**	
0000288A	0001	TimeVariantTypeEnum **- type**	0x01
0000288B	0004	signed integer **- groupID**	0x00000000

Figure 93: TimePropertyList4TimeNodeContainer record B child-record hierarchy in the fifth-level ExtTimeNodeContainer

case of TL_TPID_EffectNodeType.effectNodeType: 0x00000001 specifies that this time node is for a click effect.

case of TL_TPID_EffectID.effectID: 0x00000001 specifies the identifier of the animation effect applied to the paragraphs, which specifies the appear animation effect.

case of TL_TPID_EffectDir.intValue: 0x00000000 specifies no direction for the animation effect.

case of TL_TPID_EffectType.effectType: 0x00000001 specifies that the animation effect is an entrance effect.

case of TL_TPID_GroupID.groupID: 0x00000000 specifies the build identifier of the animation effect.

The child-record hierarchy of the **ExtTimeNodeContainer** record D from the table titled "Fifth-level ExtTimeNodeContainer child-record hierarchy" in this section is shown expanded in the following table. This is the only animation behavior of the appear animation effect.

Offset	Size	Structure	Value
000028AF	0100	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
000028AF	0008	RecordHeader **- rh**	
000028B7	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
000028DF	0008	**B**: TimePropertyList4TimeNodeContainer **- timePropertyList**	
000028DF	0008	RecordHeader **- rh**	
000028E7	00A8	**C**: TimeSetBehaviorContainer **- timeSetBehavior**	
0000298F	0020	**D**: TimeConditionContainer **- timeCondition**	
0000298F	0008	RecordHeader **- rh**	
00002997	0018	TimeConditionAtom **- conditionAtom**	
00002997	0008	RecordHeader **- rh**	
0000299F	0004	TriggerObjectEnum **- triggerObject**	0x00000000
000029A3	0004	unsigned integer **- triggerEvent**	0x00000000
000029A7	0004	unsigned integer **- id**	0x00000000
000029AB	0004	signed integer **- delay**	0x00000000

Figure 94: Sixth-level ExtTimeNodeContainer child-record hierarchy

The values of the **conditionAtom.triggerObject** field and the **conditionAtom.triggerEvent** field in the **TimeConditionContainer** record D are both 0x00000000, which specifies no condition for this time node and implies that the evaluation of the time condition always returns true.

The **TimePropertyList4TimeNodeContainer** record B is an empty list.

The child-record hierarchy of the TimeNodeAtom record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
000028B7	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
000028B7	0008	RecordHeader **- rh**	
000028BF	0004	unsigned integer **- reserved1**	0x00000000
000028C3	0004	unsigned integer **- restart**	0x00000000
000028C7	0004	TimeNodeTypeEnum **- type**	0x00000003
000028CB	0004	unsigned integer **- fill**	0x00000003
000028CF	0004	signed integer **- reserved2**	0x00000000

Offset	Size	Structure	Value
000028D3	0001	unsigned integer **- reserved3**	0x00
000028D7	0004	signed integer **- duration**	0x00000001
000028DB	1 bit	bit **- fFillProperty**	0x1
000028DB	1 bit	bit **- fRestartProperty**	0x0
000028DB	1 bit	bit **- reserved4**	0x0
000028DB	1 bit	bit **- fGroupingTypeProperty**	0x1
000028DB	1 bit	bit **- fDurationProperty**	0x1
000028DB	27 bits	unsigned integer **- reserved5**	0x0000000

Figure 95: TimeNodeAtom record A child-record hierarchy in the sixth-level ExtTimeNodeContainer

type: 0x00000003 specifies that this time node is a behavior time node that contains a behavior.

fill: 0x00000003 specifies that the properties remain at their ending values while the parent time node at the fourth level is still running or holding.

duration: 0x00000001 specifies that the duration of this time node is 0.001 seconds.

The child-record hierarchy of the **TimeSetBehaviorContainer** (section 2.8.69) record C from the table titled "Sixth-level ExtTimeNodeContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
000028E7	00A8	TimeSetBehaviorContainer **- timeSetBehavior**	
000028E7	0008	RecordHeader **- rh**	
000028EF	0010	TimeSetBehaviorAtom **- setBehaviorAtom**	
000028EF	0008	RecordHeader **- rh**	
000028EF	1 bit	bit **- fToPropertyUsed**	0x1
000028EF	1 bit	bit **- fValueTypePropertyUsed**	0x0
000028EF	30 bits	unsigned integer **- reserved**	0x00000000
000028FB	0004	TimeAnimateBehaviorValueTypeEnum **- valueType**	0x00000001
000028FF	0019	TimeVariantString **- varTo**	
000028FF	0008	RecordHeader **- rh**	
00002907	0001	TimeVariantTypeEnum **- type**	0x03
00002908	0010	array of bytes **- stringValue**	visible
00002918	0077	**A**: TimeBehaviorContainer **- behavior**	

Figure 96: Child-record hierarchy of TimeSetBehaviorContainer in the sixth-level ExtTimeNodeContainer

This set behavior assigns a specified value to the property as specified in the **timeSetBehavior.varTo.stringValue** field.

setBehaviorAtom.valueType: 0x00000001 specifies that the type of the property value is a number.

varTo.stringValue: "visible" specifies the preset value that will be assigned to the property that controls the visibility of the target object. After this value is set, the target object appears in the slide show.

The child-record hierarchy of the **TimeBehaviorContainer** (section 2.8.34) record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00002918	0077	TimeBehaviorContainer **- behavior**	
00002918	0008	RecordHeader **- rh**	
00002920	0018	**A**: TimeBehaviorAtom **- behaviorAtom**	
00002938	0033	**B**: TimeStringListContainer **- stringList**	
00002938	0008	RecordHeader **- rh**	
00002940	002B	TimeVariantString **- string**	
00002940	0008	RecordHeader **- rh**	
00002948	0001	TimeVariantTypeEnum **- type**	0x03
00002949	0022	array of bytes **- stringValue**	style.visibility
0000296B	0024	**C**: ClientVisualElementContainer **- clientVisualElement**	
0000296B	0008	RecordHeader **- rh**	
00002973	001C	VisualElementAtom **- visualElementAtom**	
00002973	001C	VisualShapeOrSoundAtom **- case of RT_VisualShapeAtom**	
00002973	001C	VisualShapeAtom **- case of TL_ET_ShapeType**	
00002973	001C	VisualShapeGeneralAtom **- default_case**	
00002973	0008	RecordHeader **- rh**	
0000297B	0004	TimeVisualElementEnum **- type**	0x00000002
0000297F	0004	ElementTypeEnum **- refType**	0x00000001
00002983	0004	unsigned integer **- shapeIdRef**	0x00000C03
00002987	0004	signed integer **- data1**	0x00000000
0000298B	0004	signed integer **- data2**	0x0000000C

Figure 97: Child-record hierarchy of TimeBehaviorContainer

stringList.string.stringValue: "style.visibility" specifies the property to which a value is assigned. This property controls the visibility of the target object.

clientVisualElement.visualElementAtom.case of RT_VisualShapeAtom.case of TL_ET_ShapeType.default_case.type: 0x00000002 specifies that the animation is applied to a specified range of text in the shape.

clientVisualElement.visualElementAtom.case of RT_VisualShapeAtom.case of TL_ET_ShapeType.default_case.refType: 0x00000001 specifies that the animation target is a part of the shape that contains the three paragraphs.

clientVisualElement.visualElementAtom.case of RT_VisualShapeAtom.case of TL_ET_ShapeType.default_case.shapeIdRef: 0x00000C03 specifies the **shape identifier** of the shape that contains the three paragraphs.

clientVisualElement.visualElementAtom.case of RT_VisualShapeAtom.case of TL_ET_ShapeType.default_case.data1: 0x00000000 specifies the character index of the beginning of the paragraph to be animated.

clientVisualElement.visualElementAtom.case of RT_VisualShapeAtom.case of TL_ET_ShapeType.default_case.data2: 0x0000000C specifies the character index of the end of the paragraph to be animated.

The child-record hierarchy of the TimeBehaviorAtom record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00002920	0018	**A**: TimeBehaviorAtom **- behaviorAtom**	
00002920	0008	RecordHeader **- rh**	
00002920	1 bit	bit **- fAdditivePropertyUsed**	0x0
00002920	1 bit	bit **- reserved1**	0x0
00002920	1 bit	bit **- fAttributeNamesPropertyUsed**	0x1
00002920	1 bit	bit **- reserved2**	0x0
00002920	28 bits	unsigned integer **- reserved3**	0x0000000
0000292C	0004	unsigned integer **- behaviorAdditive**	0x00000000
00002930	0004	signed integer **- behaviorAccumulate**	0x00000000
00002934	0004	signed integer **- behaviorTransform**	0x00000000

Figure 98: TimeBehaviorAtom record A child-record hierarchy

behaviorAdditive: 0x00000000 specifies that the animated value replaces the original value of the property to be animated.

behaviorAccumulate: 0x00000000 specifies that no accumulation between repeating animations occurs.

behaviorTransform: 0x00000000 specifies that the animation animates a property of the target object.

3.7.2 Shape Animation Example

This example explains the timing tree of the shape animations for the green square on presentation slide 3 as shown in figure titled "Presentation slide 3" in section 3.1. The effects include a fly-in

animation effect, a spin animation effect, and a sink-down animation effect. These three animation effects are triggered one after the other by successive mouse clicks.

The timing tree of the shape animation is found inside the PP2.5.24SlideBinaryTagExtension record. For example, the **ExtTimeNodeContainer** (section 2.8.15) record D from the table titled "SlideProgTagsContainer child-record hierarchy" in section 3.6.2 is the root of the timing tree for presentation slide 2. The **ExtTimeNodeContainer** record shown expanded in the following table is similarly the root of the timing tree for presentation slide 3.

Offset	Size	Structure	Value
00006C0E	0E38	ExtTimeNodeContainer **- extTimeNodeContainer**	
00006C0E	0008	RecordHeader **- rh**	
00006C16	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
00006C3E	0015	TimePropertyList4TimeNodeContainer **- timePropertyList**	
00006C3E	0008	RecordHeader **- rh**	
00006C46	000D	TimeVariant4TimeNode **- rec**	
00006C46	000D	TimeEffectNodeType **- case of TL_TPID_EffectNodeType**	
00006C46	000D	TimeVariantInt **- effectNodeType0**	
00006C46	0008	RecordHeader **- rh**	
00006C4E	0001	TimeVariantTypeEnum **- type**	0x01
00006C4F	0004	signed integer **- effectNodeType**	0x00000009
00006C53	0DF3	**B**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	

Figure 99: First-level ExtTimeNodeContainer child-record hierarchy

timePropertyList.rec.case of TL_TPID_EffectNodeType.effectNodeType0.effectNodeType: 0x00000009 specifies that this time node is the root of the timing tree.

The TimeNodeAtom record A is similar to the TimeNodeAtom record, as specified in the table titled "TimeNodeAtom record A child-record hierarchy" in section 3.7.1.

The child-record hierarchy of the **ExtTimeNodeContainer** record B from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00006C53	0DF3	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
00006C53	0008	RecordHeader **- rh**	
00006C5B	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
00006C83	0015	TimePropertyList4TimeNodeContainer **- timePropertyList**	
00006C83	0008	RecordHeader **- rh**	
00006C8B	000D	TimeEffectNodeType **- case of TL_TPID_EffectNodeType**	
00006C8B	0008	RecordHeader **- rh**	

Offset	Size	Structure	Value
00006C93	0001	TimeVariantTypeEnum **- type**	0x01
00006C94	0004	signed integer **- effectNodeType**	0x00000004
00006C98	001C	**B**: TimeSequenceDataAtom **- timeSequenceDataAtom**	
00006CB4	0034	**C**: TimeConditionContainer **- timeCondition**	
00006CE8	0034	**D**: TimeConditionContainer **- timeCondition**	
00006D1C	0483	**E**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
0000719F	022E	**F**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
000073CD	0679	**G**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	

Figure 100: Second-level ExtTimeNodeContainer child-record hierarchy

As a second-level time node, the **ExtTimeNodeContainer** record shown in the preceding table serves a similar purpose in the timing tree as does the **ExtTimeNodeContainer** record shown in the table titled "Second-level ExtTimeNodeContainer child-record hierarchy" in this section. Consequently many of the child-records are likewise similar. The TimeNodeAtom record A specifies the timing attributes of this time node and is similar to the TimeNodeAtom record, as specified in the table titled "TimeNodeAtom record A child-record hierarchy in the second-level ExtTimeNodeContainer" in section 3.7.1. The TimeSequenceDataAtom record B specifies how to activate the child time nodes E, F, and G, sequentially and is similar to the TimeSequenceDataAtom record, as specified in the table titled "TimeSequenceDataAtom record B child-record hierarchy" in section 3.7.1. The **TimeConditionContainer** (section 2.8.75) record C specifies the time condition to activate the next child time node and is similar to the **TimeConditionContainer** record, as specified in the table titled "TimeConditionContainer record C child-record hierarchy in the second-level ExtTimeNodeContainer" in section 3.7.1. The **TimeConditionContainer** record D specifies the time condition required to activate the previous child time node and is similar to the **TimeConditionContainer** record as specified in the table titled "TimeConditionContainer record D child-record hierarchy in the second-level ExtTimeNodeContainer" in section 3.7.1.

timePropertyList.case of TL_TPID_EffectNodeType.effectNodeType: 0x00000004 specifies that this time node is the main sequential time node.

The child-record hierarchy of the **ExtTimeNodeContainer** record E from the previous table is shown expanded in the following table. This time node contains the fly-in animation.

Offset	Size	Structure	Value
00006D1C	0483	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
00006D1C	0008	RecordHeader **- rh**	
00006D24	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
00006D4C	0008	**B**: TimePropertyList4TimeNodeContainer **- timePropertyList**	
00006D4C	0008	RecordHeader **- rh**	
00006D54	0020	**C**: TimeConditionContainer **- timeCondition**	
00006D54	0008	RecordHeader **- rh**	
00006D5C	0018	TimeConditionAtom **- conditionAtom**	

Offset	Size	Structure	Value
00006D5C	0008	RecordHeader **- rh**	
00006D64	0004	TriggerObjectEnum **- triggerObject**	0x00000000
00006D68	0004	unsigned integer **- triggerEvent**	0x00000000
00006D6C	0004	unsigned integer **- id**	0x00000000
00006D70	0004	signed integer **- delay**	0xFFFFFFFF
00006D74	042B	**D**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	

Figure 101: Third-level ExtTimeNodeContainer child-record hierarchy

The TimeNodeAtom record A specifies the timing attributes of this time node and is similar to the TimeNodeAtom record, as specified in the table titled "TimeNodeAtom record A child-record hierarchy in the third-level ExtTimeNodeContainer" in section 3.7.1. The **TimePropertyList4TimeNodeContainer** (section 2.8.18) record B is an empty list.

The value of the **conditionAtom.delay** field in the **TimeConditionContainer** record C is 0xFFFFFFFF, which specifies an infinite value and implies that no delay is defined. The value of the **conditionAtom.triggerObject** field and the value of the **conditionAtom.triggerEvent** field in the **TimeConditionContainer** record C are both 0x00000000, which specifies no time condition for this time node and implies that the evaluation of the time condition always returns true.

The child-record hierarchy of the **ExtTimeNodeContainer** record D from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00006D74	042B	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
00006D74	0008	RecordHeader **- rh**	
00006D7C	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
00006DA4	0008	TimePropertyList4TimeNodeContainer **- timePropertyList**	
00006DA4	0008	RecordHeader **- rh**	
00006DAC	0020	TimeConditionContainer **- timeCondition**	
00006DAC	0008	RecordHeader **- rh**	
00006DB4	0018	TimeConditionAtom **- conditionAtom**	
00006DB4	0008	RecordHeader **- rh**	
00006DBC	0004	TriggerObjectEnum **- triggerObject**	0x00000000
00006DC0	0004	unsigned integer **- triggerEvent**	0x00000000
00006DC4	0004	unsigned integer **- id**	0x00000000
00006DC8	0004	signed integer **- delay**	0x00000000
00006DCC	03D3	**B**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	

Figure 102: Fourth-level ExtTimeNodeContainer child-record hierarchy

The TimeNodeAtom record A is similar to the TimeNodeAtom record A, as specified in the table titled "TimeNodeAtom record A child-record hierarchy in the fourth-level ExtTimeNodeContainer" in section 3.7.1.

The child-record hierarchy of the **ExtTimeNodeContainer** record B from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00006DCC	03D3	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
00006DCC	0008	RecordHeader **- rh**	
00006DD4	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
00006DFC	0049	**B**: TimePropertyList4TimeNodeContainer **- timePropertyList**	
00006E45	0020	TimeConditionContainer **- timeCondition**	
00006E45	0008	RecordHeader **- rh**	
00006E4D	0018	TimeConditionAtom **- conditionAtom**	
00006E4D	0008	RecordHeader **- rh**	
00006E55	0004	TriggerObjectEnum **- triggerObject**	0x00000000
00006E59	0004	unsigned integer **- triggerEvent**	0x00000000
00006E5D	0004	unsigned integer **- id**	0x00000000
00006E61	0004	signed integer **- delay**	0x00000000
00006E65	0100	**C**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
00006F65	0119	**D**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
0000707E	0121	**E**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	

Figure 103: Fifth-level ExtTimeNodeContainer child-record hierarchy

The **ExtTimeNodeContainer** record C contains a set behavior, while the **ExtTimeNodeContainer** records D and E contain two generic animations. These three animation behaviors form the fly-in animation that is triggered by the first mouse click.

The child-record hierarchy of the TimeNodeAtom record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00006DD4	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
00006DD4	0008	RecordHeader **- rh**	
00006DDC	0004	unsigned integer **- reserved1**	0x00000000
00006DE0	0004	unsigned integer **- restart**	0x00000000
00006DE4	0004	TimeNodeTypeEnum **- type**	0x00000000
00006DE8	0004	unsigned integer **- fill**	0x00000003

Offset	Size	Structure	Value
00006DEC	0004	signed integer **- reserved2**	0x00000000
00006DF0	0001	unsigned integer **- reserved3**	0x00
00006DF4	0004	signed integer **- duration**	0x00000000
00006DF8	1 bit	bit **- fFillProperty**	0x1
00006DF8	1 bit	bit **- fRestartProperty**	0x0
00006DF8	1 bit	bit **- reserved4**	0x0
00006DF8	1 bit	bit **- fGroupingTypeProperty**	0x0
00006DF8	1 bit	bit **- fDurationProperty**	0x0
00006DF8	27 bits	unsigned integer **- reserved5**	0x0000000

Figure 104: TimeNodeAtom record A child-record hierarchy in the fifth-level ExtTimeNodeContainer

fill: 0x00000003 specifies that the properties remain at their ending values while the parent time node at the fourth level is still running or holding.

duration: 0x00000000 specifies that the duration of this time node depends on its child nodes.

The child-record hierarchy of the **TimePropertyList2.8.18TimeNodeContainer** record B from the first table titled "Fifth-level ExtTimeNodeContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00006DFC	0049	**B**: TimePropertyList4TimeNodeContainer **- timePropertyList**	
00006DFC	0008	RecordHeader **- rh**	
00006E04	000D	TimeEffectNodeType **- case of TL_TPID_EffectNodeType**	
00006E04	0008	RecordHeader **- rh**	
00006E0C	0001	TimeVariantTypeEnum **- type**	0x01
00006E0D	0004	signed integer **- effectNodeType**	0x00000001
00006E11	000D	TimeEffectID **- case of TL_TPID_EffectID**	
00006E11	0008	RecordHeader **- rh**	
00006E19	0001	TimeVariantTypeEnum **- type**	0x01
00006E1A	0004	signed integer **- effectID**	0x00000002
00006E1E	000D	TimeVariantInt **- case of TL_TPID_EffectDir**	
00006E26	0001	TimeVariantTypeEnum **- type**	0x01
00006E27	0004	signed integer **- intValue**	0x00000004
00006E2B	000D	TimeEffectType **- case of TL_TPID_EffectType**	
00006E2B	0008	RecordHeader **- rh**	

Offset	Size	Structure	Value
00006E33	0001	TimeVariantTypeEnum **- type**	0x01
00006E34	0004	signed integer **- effectType**	0x00000001
00006E38	000D	TimeGroupID **- case of TL_TPID_GroupID**	
00006E38	0008	RecordHeader **- rh**	
00006E40	0001	TimeVariantTypeEnum **- type**	0x01
00006E41	0004	signed integer **- groupID**	0x00000001

Figure 105: Child-record hierarchy of TimePropertyList4TimeNodeContainer in the fifth-level ExtTimeNodeContainer

case of TL_TPID_EffectNodeType.effectNodeType: 0x00000001 specifies that this time node is for a click effect.

case of TL_TPID_EffectID.effectID: 0x00000002 specifies the identifier of the fly-in animation effect.

case of TL_TPID_EffectDir.intValue: 0x00000004 specifies that the direction of the fly-in animation is from bottom up.

case of TL_TPID_EffectType.effectType: 0x00000001 specifies that the fly-in animation is an entrance effect.

case of TL_TPID_GroupID.groupID: 0x00000001 specifies the build identifier of the fly-in animation.

The child-record hierarchy of the **ExtTimeNodeContainer** record C from the first table titled "Fifth-level ExtTimeNodeContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00006E65	0100	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
00006E65	0008	RecordHeader **- rh**	
00006E6D	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
00006E95	0008	TimePropertyList4TimeNodeContainer **- timePropertyList**	
00006E95	0008	RecordHeader **- rh**	
00006E9D	00A8	**B**: TimeSetBehaviorContainer **- timeSetBehavior**	
00006F45	0020	TimeConditionContainer **- timeCondition**	
00006F45	0008	RecordHeader **- rh**	
00006F4D	0018	TimeConditionAtom **- conditionAtom**	
00006F4D	0008	RecordHeader **- rh**	
00006F55	0004	TriggerObjectEnum **- triggerObject**	0x00000000
00006F59	0004	unsigned integer **- triggerEvent**	0x00000000
00006F5D	0004	unsigned integer **- id**	0x00000000

Offset	Size	Structure	Value
00006F61	0004	signed integer **- delay**	0x00000000

Figure 106: The first sixth-level ExtTimeNodeContainer child-record hierarchy

This time node contains a set behavior that sets the "style.visibility" property of the shape to "visible", to make the shape visible in the slide show. It is very similar to the time node specified in the table titled "Sixth-level ExtTimeNodeContainer child-record hierarchy" in section 3.7.1. The TimeNodeAtom record A is similar to the TimeNodeAtom record specified in the table titled "TimeNodeAtom record A child-record hierarchy in the sixth-level ExtTimeNodeContainer" in section 3.7.1, while the **TimeSetBehaviorContainer** (section 2.8.69) record B is similar to the **TimeSetBehaviorContainer** specified in the table titled "Child-record hierarchy of TimeSetBehaviorContainer in the sixth-level ExtTimeNodeContainer" in section 3.7.1. The **timePropertyList** field is an empty list, and the **timeCondition** field specifies that the evaluation of the time condition always returns true.

The child-record hierarchy of the **ExtTimeNodeContainer** record D from the first table titled "Fifth-level ExtTimeNodeContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00006F65	0119	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
00006F65	0008	RecordHeader **- rh**	
00006F6D	0028	TimeNodeAtom **- timeNodeAtom**	
00006F6D	0008	RecordHeader **- rh**	
00006F75	0004	unsigned integer **- reserved1**	0x00000000
00006F79	0004	unsigned integer **- restart**	0x00000000
00006F7D	0004	TimeNodeTypeEnum **- type**	0x00000003
00006F81	0004	unsigned integer **- fill**	0x00000003
00006F85	0004	signed integer **- reserved2**	0x00000000
00006F89	0001	unsigned integer **- reserved3**	0x00
00006F8D	0004	signed integer **- duration**	0x000001F4
00006F91	1 bit	bit **- fFillProperty**	0x1
00006F91	1 bit	bit **- fRestartProperty**	0x0
00006F91	1 bit	bit **- reserved4**	0x0
00006F91	1 bit	bit **- fGroupingTypeProperty**	0x1
00006F91	1 bit	bit **- fDurationProperty**	0x1
00006F91	27 bits	unsigned integer **- reserved5**	0x0000000
00006F95	0008	TimePropertyList4TimeNodeContainer **- timePropertyList**	
00006F95	0008	RecordHeader **- rh**	
00006F9D	00E1	**A**: TimeAnimateBehaviorContainer **- timeAnimateBehavior**	

Figure 107: The second sixth-level ExtTimeNodeContainer child-record hierarchy

This time node contains a generic behavior that animates the position of the shape in the horizontal axis.

timeNodeAtom.type: 0x00000003 specifies that this time node is a behavior time node that contains a behavior.

timeNodeAtom.fill: 0x00000003 specifies that the properties remain at their ending values while the parent time node at the fourth level is still running or holding.

timeNodeAtom.duration: 0x000001F4 specifies that the duration of this time node is 0.5 seconds.

The child-record hierarchy of the **TimeAnimateBehaviorContainer** (section 2.8.29) record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00006F9D	00E1	TimeAnimateBehaviorContainer **- timeAnimateBehavior**	
00006F9D	0008	RecordHeader **- rh**	
00006FA5	0014	TimeAnimateBehaviorAtom **- animateBehaviorAtom**	
00006FA5	0008	RecordHeader **- rh**	
00006FAD	0004	unsigned integer **- calcMode**	0x00000001
00006FB1	1 bit	bit **- fByPropertyUsed**	0x0
00006FB1	1 bit	bit **- fFromPropertyUsed**	0x0
00006FB1	1 bit	bit **- fToPropertyUsed**	0x0
00006FB1	1 bit	bit **- fCalcModePropertyUsed**	0x1
00006FB1	1 bit	bit **- fAnimationValuesPropertyUsed**	0x1
00006FB1	1 bit	bit **- fValueTypePropertyUsed**	0x1
00006FB1	26 bits	unsigned integer **- reserved**	0x0000000
00006FB5	0004	TimeAnimateBehaviorValueTypeEnum **- valueType**	0x00000001
00006FB9	0064	**A**: TimeAnimationValueListContainer **- animateValueList**	
0000701D	0061	**B**: TimeBehaviorContainer **- behavior**	

Figure 108: TimeAnimateBehaviorContainer child-record hierarchy in the sixth-level ExtTimeNodeContainer

animateBehaviorAtom.calcMode: 0x00000001 specifies that the animated value of the property is calculated by linear interpolation.

animateBehaviorAtom.valueType: 0x00000001 specifies that the property value is a number.

The child-record hierarchy of the **TimeAnimationValueListContainer** (section 2.8.31) record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00006FB9	0064	TimeAnimationValueListContainer **- animateValueList**	
00006FB9	0008	RecordHeader **- rh**	

Offset	Size	Structure	Value
00006FC1	002E	**A**: TimeAnimationValueListEntry **- timeAnimValueListEntry[0]**	
00006FC1	000C	TimeAnimationValueAtom **- timeAnimationValueAtom**	
00006FC1	0008	RecordHeader **- rh**	
00006FC9	0004	signed integer **- time**	0x00000000
00006FCD	0017	TimeVariantString **- case of TL_TVT_String**	
00006FCD	0008	RecordHeader **- rh**	
00006FD5	0001	TimeVariantTypeEnum **- type**	0x03
00006FD6	000E	array of bytes **- stringValue**	#ppt_x
00006FE4	000B	TimeVariantString **- varFormula**	
00006FE4	0008	RecordHeader **- rh**	
00006FEC	0001	TimeVariantTypeEnum **- type**	0x03
00006FED	0002	array of bytes **- stringValue**	
00006FEF	002E	**B**: TimeAnimationValueListEntry **- timeAnimValueListEntry[1]**	
00006FEF	000C	TimeAnimationValueAtom **- timeAnimationValueAtom**	
00006FEF	0008	RecordHeader **- rh**	
00006FF7	0004	signed integer **- time**	0x000003E8
00006FFB	0017	TimeVariantString **- case of TL_TVT_String**	
00006FFB	0008	RecordHeader **- rh**	
00007003	0001	TimeVariantTypeEnum **- type**	0x03
00007004	000E	array of bytes **- stringValue**	#ppt_x
00007012	000B	TimeVariantString **- varFormula**	
00007012	0008	RecordHeader **- rh**	
0000701A	0001	TimeVariantTypeEnum **- type**	0x03
0000701B	0002	array of bytes **- stringValue**	

Figure 109: TimeAnimationValueListContainer child-record hierarchy

The **TimeAnimationValueListContainer** record specifies two key points of the animation. The TimeAnimationValueListEntry record A specifies the beginning point, while the TimeAnimationValueListEntry record B specifies the endpoint.

timeAnimValueListEntry[0].timeAnimationValueAtom.time: 0x00000000 specifies that the first key point is the starting point of the animation.

timeAnimValueListEntry[0].case of TL_TVT_String.stringValue: "#ppt_x" specifies the formula that is used to calculate the property value at time 0x00000000. The value of this formula is the original value of the position of the shape on the horizontal axis.

timeAnimValueListEntry[1].timeAnimationValueAtom.time: 0x000003E8 specifies that the second key point is the endpoint of the animation.

timeAnimValueListEntry[1].case of TL_TVT_String.stringValue: "#ppt_x" specifies the formula that is used to calculate the property value at time 0x000003E8. The value of this formula is the original value of the position of the shape on the horizontal axis.

Because the values at the two key points are identical and equal to the original value of the position on the horizontal axis, the shape does not change its position on the horizontal axis during the fly-in animation.

The child-record hierarchy of the **TimeBehaviorContainer** (section 2.8.34) record B from the first table titled "TimeAnimateBehaviorContainer child-record hierarchy in the sixth-level ExtTimeNodeContainer" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
0000701D	0061	TimeBehaviorContainer **- behavior**	
0000701D	0008	RecordHeader **- rh**	
00007025	0018	**A**: TimeBehaviorAtom **- behaviorAtom**	
0000703D	001D	**B**: TimeStringListContainer **- stringList**	
0000703D	0008	RecordHeader **- rh**	
00007045	0015	TimeVariantString **- string**	
00007045	0008	RecordHeader **- rh**	
0000704D	0001	TimeVariantTypeEnum **- type**	0x03
0000704E	000C	array of bytes **- stringValue**	ppt_x
0000705A	0024	**C**: ClientVisualElementContainer **- clientVisualElement**	
0000705A	0008	RecordHeader **- rh**	
00007062	001C	VisualElementAtom **- visualElementAtom**	
00007062	001C	VisualShapeOrSoundAtom **- case of RT_VisualShapeAtom**	
00007062	001C	VisualShapeAtom **- case of TL_ET_ShapeType**	
00007062	001C	VisualShapeGeneralAtom **- default_case**	
00007062	0008	RecordHeader **- rh**	
0000706A	0004	TimeVisualElementEnum **- type**	0x00000000
0000706E	0004	ElementTypeEnum **- refType**	0x00000001
00007072	0004	unsigned integer **- shapeIdRef**	0x00001404
00007076	0004	signed integer **- data1**	0xFFFFFFFF
0000707A	0004	signed integer **- data2**	0xFFFFFFFF

Figure 110: TimeBehaviorContainer child-record hierarchy

The **TimeStringListContainer** (section 2.8.36) record B specifies the properties to be animated.

stringList.string.stringValue: "ppt_x" specifies that the property to be animated is the position of the shape on the horizontal axis.

clientVisualElement.visualElementAtom.case of RT_VisualShapeAtom.case of TL_ET_ShapeType.default_case.type: 0x00000000 specifies that the animation is applied to the shape.

clientVisualElement.visualElementAtom.case of RT_VisualShapeAtom.case of TL_ET_ShapeType.default_case.refType: 0x00000001 specifies that the animation target is the shape.

clientVisualElement.visualElementAtom.case of RT_VisualShapeAtom.case of TL_ET_ShapeType.default_case.shapeIdRef: 0x00001404 specifies the shape identifier of the target shape.

The child-record hierarchy of the TimeBehaviorAtom record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00007025	0018	**A**: TimeBehaviorAtom **- behaviorAtom**	
00007025	0008	RecordHeader **- rh**	
00007025	1 bit	bit **- fAdditivePropertyUsed**	0x1
00007025	1 bit	bit **- reserved1**	0x0
00007025	1 bit	bit **- fAttributeNamesPropertyUsed**	0x1
00007025	1 bit	bit **- reserved2**	0x0
00007025	28 bits	unsigned integer **- reserved3**	0x0000000
00007031	0004	unsigned integer **- behaviorAdditive**	0x00000000
00007035	0004	signed integer **- behaviorAccumulate**	0x00000000
00007039	0004	signed integer **- behaviorTransform**	0x00000000

Figure 111: TimeBehaviorAtom record A child-record hierarchy

behaviorAdditive: 0x00000000 specifies that the animated value replaces the original value of the property being animated.

behaviorAccumulate: 0x00000000 specifies that no accumulation between repeating animations occurs.

behaviorTransform: 0x00000000 specifies that the animation animates a property of the target object.

The child-record hierarchy of the **ExtTimeNodeContainer** record E from the first table titled "Fifth-level ExtTimeNodeContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
0000707E	0121	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
0000707E	0008	RecordHeader **- rh**	
00007086	0028	TimeNodeAtom **- timeNodeAtom**	

Offset	Size	Structure	Value
00007086	0008	RecordHeader **- rh**	
0000708E	0004	unsigned integer **- reserved1**	0x00000000
00007092	0004	unsigned integer **- restart**	0x00000000
00007096	0004	TimeNodeTypeEnum **- type**	0x00000003
0000709A	0004	unsigned integer **- fill**	0x00000003
0000709E	0004	signed integer **- reserved2**	0x00000000
000070A2	0001	unsigned integer **- reserved3**	0x00
000070A6	0004	signed integer **- duration**	0x000001F4
000070AA	1 bit	bit **- fFillProperty**	0x1
000070AA	1 bit	bit **- fRestartProperty**	0x0
000070AA	1 bit	bit **- reserved4**	0x0
000070AA	1 bit	bit **- fGroupingTypeProperty**	0x1
000070AA	1 bit	bit **- fDurationProperty**	0x1
000070AA	27 bits	unsigned integer **- reserved5**	0x0000000
000070AE	0008	TimePropertyList4TimeNodeContainer **- timePropertyList**	
000070AE	0008	RecordHeader **- rh**	
000070B6	00E9	**A**: TimeAnimateBehaviorContainer **- timeAnimateBehavior**	

Figure 112: The third sixth-level ExtTimeNodeContainer child-record hierarchy

This time node contains a generic behavior that animates the position of the shape on the vertical axis.

timeNodeAtom.duration: 0x000001F4 specifies that the duration of this time node is 0.5 seconds.

The child-record hierarchy of the **TimeAnimateBehaviorContainer** (section 2.8.29) record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
000070B6	00E9	TimeAnimateBehaviorContainer **- timeAnimateBehavior**	
000070B6	0008	RecordHeader **- rh**	
000070BE	0014	TimeAnimateBehaviorAtom **- animateBehaviorAtom**	
000070BE	0008	RecordHeader **- rh**	
000070C6	0004	unsigned integer **- calcMode**	0x00000001
000070CA	0004	unsigned integer **- fByPropertyUsed**	0x00000000
000070CA	0004	unsigned integer **- fFromPropertyUsed**	0x00000000
000070CA	0004	unsigned integer **- fToPropertyUsed**	0x00000000

Offset	Size	Structure	Value
000070CA	0004	unsigned integer **- fCalcModePropertyUsed**	0x00000001
000070CA	0004	unsigned integer **- fAnimationValuesPropertyUsed**	0x00000001
000070CA	0004	unsigned integer **- fValueTypePropertyUsed**	0x00000001
000070CA	0004	unsigned integer **- reserved**	0x00000000
000070CE	0004	TimeAnimateBehaviorValueTypeEnum **- valueType**	0x00000001
000070D2	006C	**A**: TimeAnimationValueListContainer **- animateValueList**	
0000713E	0061	**B**: TimeBehaviorContainer **- behavior**	

Figure 113: TimeAnimateBehaviorContainer child-record hierarchy in the sixth-level ExtTimeNodeContainer

animateBehaviorAtom.calcMode: 0x00000001 specifies that the animated value of the property is calculated by linear interpolation.

animateBehaviorAtom.valueType: 0x00000001 specifies that the property value is a number.

The child-record hierarchy of the **TimeAnimationValueListContainer** record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
000070D2	006C	TimeAnimationValueListContainer **- animateValueList**	
000070D2	0008	RecordHeader **- rh**	
000070DA	0036	**A**: TimeAnimationValueListEntry **- timeAnimValueListEntry[0]**	
000070DA	000C	TimeAnimationValueAtom **- timeAnimationValueAtom**	
000070DA	0008	RecordHeader **- rh**	
000070E2	0004	signed integer **- time**	0x00000000
000070E6	001F	TimeVariantString **- case of TL_TVT_String**	
000070E6	0008	RecordHeader **- rh**	
000070EE	0001	TimeVariantTypeEnum **- type**	0x03
000070EF	0016	array of bytes **- stringValue**	1+#ppt_h/2
00007105	000B	TimeVariantString **- varFormula**	
00007105	0008	RecordHeader **- rh**	
0000710D	0001	TimeVariantTypeEnum **- type**	0x03
0000710E	0002	array of bytes **- stringValue**	
00007110	002E	**B**: TimeAnimationValueListEntry **- timeAnimValueListEntry[1]**	
00007110	000C	TimeAnimationValueAtom **- timeAnimationValueAtom**	
00007110	0008	RecordHeader **- rh**	

Offset	Size	Structure	Value
00007118	0004	signed integer **- time**	0x000003E8
0000711C	0017	TimeVariantString **- case of TL_TVT_String**	
0000711C	0008	RecordHeader **- rh**	
00007124	0001	TimeVariantTypeEnum **- type**	0x03
00007125	000E	array of bytes **- stringValue**	#ppt_y
00007133	000B	TimeVariantString **- varFormula**	
00007133	0008	RecordHeader **- rh**	
0000713B	0001	TimeVariantTypeEnum **- type**	0x03
0000713C	0002	array of bytes **- stringValue**	

Figure 114: Child-record hierarchy of TimeAnimationValueListContainer

The **TimeAnimationValueListContainer** record specifies two key points of the animation. The TimeAnimationValueListEntry record A specifies the starting point, while the TimeAnimationValueListEntry record B specifies the endpoint.

timeAnimValueListEntry[0].timeAnimationValueAtom.time: 0x00000000 specifies that the first key point is the starting point of the animation.

timeAnimValueListEntry[0].case of TL_TVT_String.stringValue: "1+#ppt_h/2" specifies the formula that is used to calculate the property value at time 0x00000000. The value of this formula is the position that is just under the bottom of the slide show, which makes the shape hidden.

timeAnimValueListEntry[1].timeAnimationValueAtom.time: 0x000003E8 specifies that the second key point is the endpoint of the animation.

timeAnimValueListEntry[1].case of TL_TVT_String.stringValue: "#ppt_y" specifies the formula that is used to calculate the property value at time 0x000003E8. The value of this formula is the original value of the position of the shape on the vertical axis.

The two key points specify that the shape moves from the bottom of the slide show to its original position.

The child-record hierarchy of the **TimeBehaviorContainer** record B from the second table titled "TimeAnimateBehaviorContainer child-record hierarchy in the sixth-level ExtTimeNodeContainer" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
0000713E	0061	TimeBehaviorContainer **- behavior**	
0000713E	0008	RecordHeader **- rh**	
00007146	0018	**A**: TimeBehaviorAtom **- behaviorAtom**	
0000715E	001D	**B**: TimeStringListContainer **- stringList**	
0000715E	0008	RecordHeader **- rh**	
00007166	0015	TimeVariantString **- string**	
00007166	0008	RecordHeader **- rh**	

Offset	Size	Structure	Value
0000716E	0001	TimeVariantTypeEnum **- type**	0x03
0000716F	000C	array of bytes **- stringValue**	ppt_y
0000717B	0024	**C**: ClientVisualElementContainer **- clientVisualElement**	
0000717B	0008	RecordHeader **- rh**	
00007183	001C	VisualElementAtom **- visualElementAtom**	
00007183	001C	VisualShapeOrSoundAtom **- case of RT_VisualShapeAtom**	
00007183	001C	VisualShapeAtom **- case of TL_ET_ShapeType**	
00007183	001C	VisualShapeGeneralAtom **- default_case**	
00007183	0008	RecordHeader **- rh**	
0000718B	0004	TimeVisualElementEnum **- type**	0x00000000
0000718F	0004	ElementTypeEnum **- refType**	0x00000001
00007193	0004	unsigned integer **- shapeIdRef**	0x00001404
00007197	0004	signed integer **- data1**	0xFFFFFFFF
0000719B	0004	signed integer **- data2**	0xFFFFFFFF

Figure 115: Child-record hierarchy of TimeBehaviorContainer

The TimeBehaviorAtom record A is similar to the TimeBehaviorAtom record specified in the second table titled "TimeBehaviorAtom record A child-record hierarchy" in this section. The **TimeStringListContainer** record B specifies the properties to be animated. The **ClientVisualElementContainer** (section 2.8.44) record B specifies that the target object of the animation is the shape.

stringList.string.stringValue: "ppt_y" specifies that the property to be animated is the position of the shape on the vertical axis.

The child-record hierarchy of the **ExtTimeNodeContainer** record F from the table titled "Second-level ExtTimeNodeContainer child-record hierarchy" in this section is shown expanded in the following table. This time node contains the spin animation effect.

Offset	Size	Structure
0000719F	022E	**A**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**
0000719F	0008	RecordHeader **- rh**
000071A7	0028	TimeNodeAtom **- timeNodeAtom**
000071CF	0008	TimePropertyList4TimeNodeContainer **- timePropertyList**
000071D7	0020	TimeConditionContainer **- timeCondition**
000071F7	01D6	**B**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**
000071F7	0008	RecordHeader **- rh**
000071FF	0028	TimeNodeAtom **- timeNodeAtom**

Offset	Size	Structure
00007227	0008	TimePropertyList4TimeNodeContainer **- timePropertyList**
0000722F	0020	TimeConditionContainer **- timeCondition**
0000724F	017E	**C**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**

Figure 116: Third-level and fourth-level ExtTimeNodeContainer child-record hierarchies

The **ExtTimeNodeContainer** record A is similar to the **ExtTimeNodeContainer** record as specified in the second table titled "Third-level ExtTimeNodeContainer child-record hierarchy" in this section, while the **ExtTimeNodeContainer** record B is similar to the **ExtTimeNodeContainer** record as specified in the table titled "Fourth-level ExtTimeNodeContainer child-record hierarchy" in this section.

The child-record hierarchy of the **ExtTimeNodeContainer** record C from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
0000724F	017E	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
0000724F	0008	RecordHeader **- rh**	
00007257	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
0000727F	0049	**B**: TimePropertyList4TimeNodeContainer **- timePropertyList**	
000072C8	0020	TimeConditionContainer **- timeCondition**	
000072E8	0010	**C**: TimeModifierAtom **- timeModifierAtom[0]**	
000072E8	0008	RecordHeader **- rh**	
000072F0	0004	TimeModifierEnum **- type**	0x00000003
000072F4	0004	unsigned integer **- value**	0x00000000
000072F8	0010	**D**: TimeModifierAtom **- timeModifierAtom[1]**	
000072F8	0008	RecordHeader **- rh**	
00007300	0004	TimeModifierEnum **- type**	0x00000004
00007304	0004	unsigned integer **- value**	0x00000000
00007308	0010	**E**: TimeModifierAtom **- timeModifierAtom[2]**	
00007308	0008	RecordHeader **- rh**	
00007310	0004	TimeModifierEnum **- type**	0x00000005
00007314	0004	unsigned integer **- value**	0x00000000
00007318	00B5	**F**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	

Figure 117: Fifth-level ExtTimeNodeContainer child-record hierarchy

This time node specifies the spin animation effect. The TimeNodeAtom record A is similar to the TimeNodeAtom as specified in the table titled "TimeNodeAtom record A child-record hierarchy in the fifth-level ExtTimeNodeContainer" in this section. The TimeModifierAtom records C, D, and E specify the acceleration, the deceleration, and the automatic reverse attribute of this time node, respectively. The **ExtTimeNodeContainer** record F is the only child time node and contains a rotation behavior.

timeModifierAtom[0].type: 0x00000003 specifies that the **timeModifierAtom[0].value** field defines the acceleration of the spin animation effect.

timeModifierAtom[0].value: 0x00000000 specifies no acceleration of the spin animation effect.

timeModifierAtom[1].type: 0x00000004 specifies that the **timeModifierAtom[1].value** field defines the deceleration of the spin animation effect.

timeModifierAtom[1].value: 0x00000000 specifies no deceleration of the spin animation effect.

timeModifierAtom[2].type: 0x00000005 specifies that the **timeModifierAtom[2].value** field defines whether the spin animation effect automatically reverse.

timeModifierAtom[2].value: 0x00000000 specifies that the spin animation does not automatically reverse.

The child-record hierarchy of the **TimePropertyList2.8.18TimeNodeContainer** record B from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
0000727F	0049	**B**: TimePropertyList4TimeNodeContainer **- timePropertyList**	
0000727F	0008	RecordHeader **- rh**	
00007287	000D	TimeEffectNodeType **- case of TL_TPID_EffectNodeType**	
00007287	0008	RecordHeader **- rh**	
0000728F	0001	TimeVariantTypeEnum **- type**	0x01
00007290	0004	signed integer **- effectNodeType**	0x00000001
00007294	000D	TimeEffectID **- case of TL_TPID_EffectID**	
00007294	0008	RecordHeader **- rh**	
0000729C	0001	TimeVariantTypeEnum **- type**	0x01
0000729D	0004	signed integer **- effectID**	0x00000008
000072A1	000D	TimeVariantInt **- case of TL_TPID_EffectDir**	
000072A1	0008	RecordHeader **- rh**	
000072A9	0001	TimeVariantTypeEnum **- type**	0x01
000072AA	0004	signed integer **- intValue**	0x00000000
000072AE	000D	TimeEffectType **- case of TL_TPID_EffectType**	
000072AE	0008	RecordHeader **- rh**	
000072B6	0001	TimeVariantTypeEnum **- type**	0x01
000072B7	0004	signed integer **- effectType**	0x00000003
000072BB	000D	TimeGroupID **- case of TL_TPID_GroupID**	
000072BB	0008	RecordHeader **- rh**	
000072C3	0001	TimeVariantTypeEnum **- type**	0x01

Offset	Size	Structure	Value
000072C4	0004	signed integer **- groupID**	0x00000000

Figure 118: TimePropertyList4TimeNodeContainer record B child-record hierarchy in the fifth-level ExtTimeNodeContainer

case of TL_TPID_EffectNodeType.effectNodeType: 0x00000001 specifies that this time node is for a click effect.

case of TL_TPID_EffectID.effectID: 0x00000008 specifies the identifier of the spin animation effect.

case of TL_TPID_EffectDir.intValue: 0x00000000 specifies that the direction of the spin animation is clockwise.

case of TL_TPID_EffectType.effectType: 0x00000003 specifies that the spin animation effect is an emphasis effect.

case of TL_TPID_GroupID.groupID: 0x00000000 specifies the build identifier for this spin animation effect.

The child-record hierarchy of the **ExtTimeNodeContainer** record F from the second table titled "Fifth-level ExtTimeNodeContainer child-record hierarchy" in this section is shown expanded in the following table. It contains the rotation behavior that specifies how the shape spins.

Offset	Size	Structure	Value
00007318	00B5	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
00007318	0008	RecordHeader **- rh**	
00007320	0028	TimeNodeAtom **- timeNodeAtom**	
00007320	0008	RecordHeader **- rh**	
00007328	0004	unsigned integer **- reserved1**	0x00000000
0000732C	0004	unsigned integer **- restart**	0x00000000
00007330	0004	TimeNodeTypeEnum **- type**	0x00000003
00007334	0004	unsigned integer **- fill**	0x00000003
00007338	0004	signed integer **- reserved2**	0x00000000
0000733C	0001	unsigned integer **- reserved3**	0x00
00007340	0004	signed integer **- duration**	0x000007D0
00007344	1 bit	bit **- fFillProperty**	0x1
00007344	1 bit	bit **- fRestartProperty**	0x0
00007344	1 bit	bit **- reserved4**	0x0
00007344	1 bit	bit **- fGroupingTypeProperty**	0x1
00007344	1 bit	bit **- fDurationProperty**	0x1
00007344	27 bits	unsigned integer **- reserved5**	0x0000000
00007348	0008	TimePropertyList4TimeNodeContainer **- timePropertyList**	

Offset	Size	Structure	Value
00007348	0008	RecordHeader **- rh**	
00007350	007D	**A**: TimeRotationBehaviorContainer **- timeRotationBehavior**	

Figure 119: Sixth-level ExtTimeNodeContainer child-record hierarchy

timeNodeAtom.duration: 0x000007D0 specifies that the duration of the animation is 2.0 seconds.

The child-record hierarchy of the **TimeRotationBehaviorContainer** (section 2.8.65) record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00007350	007D	TimeRotationBehaviorContainer **- timeRotationBehavior**	
00007350	0008	RecordHeader **- rh**	
00007358	001C	TimeRotationBehaviorAtom **- rotationBehaviorAtom**	
00007358	0008	RecordHeader **- rh**	
00007358	1 bit	bit **- fByPropertyUsed**	0x1
00007358	1 bit	bit **- fFromPropertyUsed**	0x0
00007358	1 bit	bit **- fToPropertyUsed**	0x0
00007358	1 bit	bit **- fDirectionPropertyUsed**	0x0
00007358	28 bits	unsigned integer **- reserved**	0x0000000
00007364	0004	Float **- fBy**	360
00007368	0004	Float **- fFrom**	0
0000736C	0004	Float **- fTo**	360
00007370	0004	TimeRotationBehaviorDirectionEnum **- rotationDirection**	0x00000000
00007374	0059	**A**: TimeBehaviorContainer **- behavior**	

Figure 120: TimeRotationBehaviorContainer child-record hierarchy in the sixth-level ExtTimeNodeContainer

rotationBehaviorAtom.fBy: 360 specifies that the shape rotates 360 degrees.

rotationBehaviorAtom.fFrom: 0 specifies that the rotation starts from the original angle.

rotationBehaviorAtom.fTo: 360 specifies that the shape rotates 360 degrees.

rotationBehaviorAtom.rotationDirection: 0x00000000 specifies that the rotation is clockwise.

The child-record hierarchy of the **TimeBehaviorContainer** record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00007374	0059	TimeBehaviorContainer **- behavior**	
00007374	0008	RecordHeader **- rh**	

Offset	Size	Structure	Value
0000737C	0018	**A**: TimeBehaviorAtom **- behaviorAtom**	
00007394	0015	TimeStringListContainer **- stringList**	
00007394	0008	RecordHeader **- rh**	
0000739C	000D	TimeVariantString **- string**	
0000739C	0008	RecordHeader **- rh**	
000073A4	0001	TimeVariantTypeEnum **- type**	0x03
000073A5	0004	array of bytes **- stringValue**	r
000073A9	0024	**B**: ClientVisualElementContainer **- clientVisualElement**	
000073A9	0008	RecordHeader **- rh**	
000073B1	001C	VisualElementAtom **- visualElementAtom**	
000073B1	001C	VisualShapeOrSoundAtom **- case of RT_VisualShapeAtom**	
000073B1	001C	VisualShapeAtom **- case of TL_ET_ShapeType**	
000073B1	001C	VisualShapeGeneralAtom **- default_case**	
000073B1	0008	RecordHeader **- rh**	
000073B9	0004	TimeVisualElementEnum **- type**	0x00000000
000073BD	0004	ElementTypeEnum **- refType**	0x00000001
000073C1	0004	unsigned integer **- shapeIdRef**	0x00001404
000073C5	0004	signed integer **- data1**	0xFFFFFFFF
000073C9	0004	signed integer **- data2**	0xFFFFFFFF

Figure 121: TimeBehaviorContainer child-record hierarchy

The TimeBehaviorAtom record A is similar to the TimeBehaviorAtom as specified in the second table titled "TimeBehaviorAtom record A child-record hierarchy" in this section. The **ClientVisualElementContainer** record B specifies that the target is the shape, and it is similar to the **ClientVisualElementContainer** record as specified in the first table titled "TimeBehaviorContainer child-record hierarchy" in this section.

stringList.string.stringValue: "r" specifies that the property to be animated is the rotation angle of the shape.

The child-record hierarchy of the **ExtTimeNodeContainer** record G from the table titled "Second-level ExtTimeNodeContainer child-record hierarchy" in this section is shown expanded in the following table. This time node contains the sink-down animation.

Offset	Size	Structure
000073CD	0679	**A**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**
000073CD	0008	RecordHeader **- rh**
000073D5	0028	TimeNodeAtom **- timeNodeAtom**

Offset	Size	Structure
000073FD	0008	TimePropertyList4TimeNodeContainer **- timePropertyList**
00007405	0020	TimeConditionContainer **- timeCondition**
00007425	0621	**B**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**
00007425	0008	RecordHeader **- rh**
0000742D	0028	TimeNodeAtom **- timeNodeAtom**
00007455	0008	TimePropertyList4TimeNodeContainer **- timePropertyList**
0000745D	0020	TimeConditionContainer **- timeCondition**
0000747D	05C9	**C**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**

Figure 122: Third-level and fourth-level ExtTimeNodeContainer child-record hierarchies

The **ExtTimeNodeContainer** record A is similar to the **ExtTimeNodeContainer** record as specified in the second table titled "Third-level ExtTimeNodeContainer child-record hierarchy" in this section, while the **ExtTimeNodeContainer** record B is similar to the **ExtTimeNodeContainer** record as specified in the table titled "Fourth-level ExtTimeNodeContainer child-record hierarchy" in this section.

The child-record hierarchy of the **ExtTimeNodeContainer** record C from the previous table is shown expanded in the following table.

Offset	Size	Structure
0000747D	05C9	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**
0000747D	0008	RecordHeader **- rh**
00007485	0028	**A**: TimeNodeAtom **- timeNodeAtom**
000074AD	0049	**B**: TimePropertyList4TimeNodeContainer **- timePropertyList**
000074F6	0020	TimeConditionContainer **- timeCondition**
00007516	00A7	**C**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**
000075BD	0115	**D**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**
000076D2	012D	**E**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**
000077FF	0149	**F**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**
00007948	00FE	**G**: ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**

Figure 123: Fifth-level ExtTimeNodeContainer child-record hierarchy

This time node specifies the sink-down animation effect. The TimeNodeAtom record A is similar to the TimeNodeAtom as specified in the table titled "TimeNodeAtom record A child-record hierarchy in the fifth-level ExtTimeNodeContainer" in this section. The **ExtTimeNodeContainer** record C contains an effect behavior to fade the shape. The **ExtTimeNodeContainer** records C, D, E, and F contain generic behaviors to move the shape out of the slide show. The **ExtTimeNodeContainer** record G contains a set behavior to make the shape invisible at the end.

The child-record hierarchy of the **TimePropertyList2.8.18TimeNodeContainer** record B from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
000074AD	0049	**B**: TimePropertyList4TimeNodeContainer **- timePropertyList**	
000074AD	0008	RecordHeader **- rh**	
000074B5	000D	TimeEffectNodeType **- case of TL_TPID_EffectNodeType**	
000074B5	0008	RecordHeader **- rh**	
000074BD	0001	TimeVariantTypeEnum **- type**	0x01
000074BE	0004	signed integer **- effectNodeType**	0x00000001
000074C2	000D	TimeEffectID **- case of TL_TPID_EffectID**	
000074C2	0008	RecordHeader **- rh**	
000074CA	0001	TimeVariantTypeEnum **- type**	0x01
000074CB	0004	signed integer **- effectID**	0x00000025
000074CF	000D	TimeVariantInt **- case of TL_TPID_EffectDir**	
000074CF	0008	RecordHeader **- rh**	
000074D7	0001	TimeVariantTypeEnum **- type**	0x01
000074D8	0004	signed integer **- intValue**	0x00000000
000074DC	000D	TimeEffectType **- case of TL_TPID_EffectType**	
000074DC	0008	RecordHeader **- rh**	
000074E4	0001	TimeVariantTypeEnum **- type**	0x01
000074E5	0004	signed integer **- effectType**	0x00000002
000074E9	000D	TimeGroupID **- case of TL_TPID_GroupID**	
000074E9	0008	RecordHeader **- rh**	
000074F1	0001	TimeVariantTypeEnum **- type**	0x01
000074F2	0004	signed integer **- groupID**	0x00000002

Figure 124: TimePropertyList4TimeNodeContainer record B child-record hierarchy in the fifth-level ExtTimeNodeContainer

case of TL_TPID_EffectID.effectID: 0x00000025 specifies the identifier of this sink-down animation effect.

case of TL_TPID_EffectDir.intValue: 0x00000000 specifies no direction for this sink-down animation effect.

case of TL_TPID_EffectType.effectType: 0x00000002 specifies that this sink-down animation effect is an exit effect.

case of TL_TPID_GroupID.groupID: 0x00000002 specifies the build identifier of the sink-down animation effect.

The child-record hierarchy of the **ExtTimeNodeContainer** record C from the third table titled "Fifth-level ExtTimeNodeContainer child-record hierarchy" in this section is shown expanded in the following table. It contains an effect behavior to fade out the shape.

Offset	Size	Structure	Value
00007516	00A7	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
00007516	0008	RecordHeader **- rh**	
0000751E	0028	TimeNodeAtom **- timeNodeAtom**	
0000751E	0008	RecordHeader **- rh**	
00007526	0004	unsigned integer **- reserved1**	0x00000000
0000752A	0004	unsigned integer **- restart**	0x00000000
0000752E	0004	TimeNodeTypeEnum **- type**	0x00000003
00007532	0004	unsigned integer **- fill**	0x00000000
00007536	0004	signed integer **- reserved2**	0x00000000
0000753A	0001	unsigned integer **- reserved3**	0x00
0000753E	0004	signed integer **- duration**	0x000003E8
00007542	1 bit	bit **- fFillProperty**	0x0
00007542	1 bit	bit **- fRestartProperty**	0x0
00007542	1 bit	bit **- reserved4**	0x0
00007542	1 bit	bit **- fGroupingTypeProperty**	0x1
00007542	1 bit	bit **- fDurationProperty**	0x1
00007542	27 bits	unsigned integer **- reserved5**	0x0000000
00007546	0008	TimePropertyList4TimeNodeContainer **- timePropertyList**	
00007546	0008	RecordHeader **- rh**	
0000754E	006F	**A**: TimeEffectBehaviorContainer **- timeEffectBehavior**	

Figure 125: The first sixth-level ExtTimeNodeContainer child-record hierarchy

timeNodeAtom.fill: 0x00000000 specifies that the animated values of the property remain at their ending values while the parent time node at the fifth level is still running or holding.

timeNodeAtom.duration: 0x000003E8 specifies that the duration of the effect animation is 1.0 seconds.

The child-record hierarchy of the **TimeEffectBehaviorContainer** (section 2.8.61) record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
0000754E	006F	TimeEffectBehaviorContainer **- timeEffectBehavior**	
0000754E	0008	RecordHeader **- rh**	
00007556	0010	TimeEffectBehaviorAtom **- effectBehaviorAtom**	
00007556	0008	RecordHeader **- rh**	

Offset	Size	Structure	Value
00007556	1 bit	bit **- fTransitionPropertyUsed**	0x1
00007556	1 bit	bit **- fTypePropertyUsed**	0x1
00007556	1 bit	bit **- fProgressPropertyUsed**	0x0
00007556	1 bit	bit **- fRuntimeContextObsolete**	0x0
00007556	28 bits	unsigned integer **- reserved**	0x0000000
00007562	0004	unsigned integer **- effectTransition**	0x00000001
00007566	0013	TimeVariantString **- varType**	
00007566	0008	RecordHeader **- rh**	
0000756E	0001	TimeVariantTypeEnum **- type**	0x03
0000756F	000A	array of bytes **- stringValue**	fade
00007579	0044	**A**: TimeBehaviorContainer **- behavior**	

Figure 126: TimeEffectBehaviorContainer child-record hierarchy

effectBehaviorAtom.effectTransition: 0x00000001 specifies that the animation behavior fades out the shape.

varType.stringValue: "fade" specifies that the effect behavior is a fade effect.

The child-record hierarchy of the **TimeBehaviorContainer** record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00007579	0044	TimeBehaviorContainer **- behavior**	
00007579	0008	RecordHeader **- rh**	
00007581	0018	**A**: TimeBehaviorAtom **- behaviorAtom**	
00007581	0008	RecordHeader **- rh**	
00007581	0004	unsigned integer **- fAdditivePropertyUsed**	0x00000000
00007581	0004	unsigned integer **- reserved1**	0x00000000
00007581	0004	unsigned integer **- fAttributeNamesPropertyUsed**	0x00000000
00007581	0004	unsigned integer **- reserved2**	0x00000000
00007581	0004	unsigned integer **- reserved3**	0x00000000
0000758D	0004	unsigned integer **- behaviorAdditive**	0x00000000
00007591	0004	signed integer **- behaviorAccumulate**	0x00000000
00007595	0004	signed integer **- behaviorTransform**	0x00000000
00007599	0024	**B**: ClientVisualElementContainer **- clientVisualElement**	
00007599	0008	RecordHeader **- rh**	

Offset	Size	Structure	Value
000075A1	001C	VisualElementAtom **- visualElementAtom**	
000075A1	001C	VisualShapeOrSoundAtom **- case of RT_VisualShapeAtom**	
000075A1	001C	VisualShapeAtom **- case of TL_ET_ShapeType**	
000075A1	001C	VisualShapeGeneralAtom **- default_case**	
000075A1	0008	RecordHeader **- rh**	
000075A9	0004	TimeVisualElementEnum **- type**	0x00000000
000075AD	0004	ElementTypeEnum **- refType**	0x00000001
000075B1	0004	unsigned integer **- shapeIdRef**	0x00001404
000075B5	0004	signed integer **- data1**	0xFFFFFFFF
000075B9	0004	signed integer **- data2**	0xFFFFFFFF

Figure 127: TimeBehaviorContainer child-record hierarchy

The TimeBehaviorAtom record A is similar to the TimeBehaviorAtom record as specified in the second table titled "TimeBehaviorAtom record A child-record hierarchy" in this section. The **ClientVisualElementContainer** record B specifies that the target of the animation is the shape.

The child-record hierarchy of the **ExtTimeNodeContainer** record F from the third table titled "Fifth-level ExtTimeNodeContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
000075BD	0115	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
000075BD	0008	RecordHeader **- rh**	
000075C5	0028	TimeNodeAtom **- timeNodeAtom**	
000075C5	0008	RecordHeader **- rh**	
000075CD	0004	unsigned integer **- reserved1**	0x00000000
000075D1	0004	unsigned integer **- restart**	0x00000000
000075D5	0004	TimeNodeTypeEnum **- type**	0x00000003
000075D9	0004	unsigned integer **- fill**	0x00000000
000075DD	0004	signed integer **- reserved2**	0x00000000
000075E1	0001	unsigned integer **- reserved3**	0x00
000075E5	0004	signed integer **- duration**	0x000003E8
000075E9	1 bit	bit **- fFillProperty**	0x1
000075E9	1 bit	bit **- fRestartProperty**	0x0
000075E9	1 bit	bit **- reserved4**	0x0
000075E9	1 bit	bit **- fGroupingTypeProperty**	0x1

Offset	Size	Structure	Value
000075E9	1 bit	bit **- fDurationProperty**	0x1
000075E9	27 bits	unsigned integer **- reserved5**	0x0000000
000075ED	0008	TimePropertyList4TimeNodeContainer **- timePropertyList**	
000075ED	0008	RecordHeader **- rh**	
000075F5	00DD	**A**: TimeAnimateBehaviorContainer **- timeAnimateBehavior**	

Figure 128: The second sixth-level ExtTimeNodeContainer child-record hierarchy

This time node contains a generic behavior that animates the position of the shape on the horizontal axis.

timeNodeAtom.duration: 0x000003E8 specifies that the duration of this time node is 1.0 seconds.

The child-record hierarchy of the **TimeAnimateBehaviorContainer** (section 2.8.29) record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
000075F5	00DD	TimeAnimateBehaviorContainer **- timeAnimateBehavior**	
000075F5	0008	RecordHeader **- rh**	
000075FD	0014	TimeAnimateBehaviorAtom **- animateBehaviorAtom**	
000075FD	0008	RecordHeader **- rh**	
00007605	0004	unsigned integer **- calcMode**	0x00000001
00007609	0004	unsigned integer **- fByPropertyUsed**	0x00000000
00007609	0004	unsigned integer **- fFromPropertyUsed**	0x00000000
00007609	0004	unsigned integer **- fToPropertyUsed**	0x00000000
00007609	0004	unsigned integer **- fCalcModePropertyUsed**	0x00000001
00007609	0004	unsigned integer **- fAnimationValuesPropertyUsed**	0x00000001
00007609	0004	unsigned integer **- fValueTypePropertyUsed**	0x00000001
00007609	0004	unsigned integer **- reserved**	0x00000000
0000760D	0004	TimeAnimateBehaviorValueTypeEnum **- valueType**	0x00000001
00007611	0060	**A**: TimeAnimationValueListContainer **- animateValueList**	
00007671	0061	**B**: TimeBehaviorContainer **- behavior**	

Figure 129: TimeAnimateBehaviorContainer child-record hierarchy in the sixth-level ExtTimeNodeContainer

animateBehaviorAtom.calcMode: 0x00000001 specifies that the animated value of the property is calculated by linear interpolation.

The child-record hierarchy of the **TimeAnimationValueListContainer** record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00007611	0060	TimeAnimationValueListContainer **- animateValueList**	
00007611	0008	RecordHeader **- rh**	
00007619	002C	**A**: TimeAnimationValueListEntry **- timeAnimValueListEntry[0]**	
00007619	000C	TimeAnimationValueAtom **- timeAnimationValueAtom**	
00007619	0008	RecordHeader **- rh**	
00007621	0004	signed integer **- time**	0x00000000
00007625	0015	TimeVariantString **- case of TL_TVT_String**	
00007625	0008	RecordHeader **- rh**	
0000762D	0001	TimeVariantTypeEnum **- type**	0x03
0000762E	000C	array of bytes **- stringValue**	ppt_x
0000763A	000B	TimeVariantString **- varFormula**	
0000763A	0008	RecordHeader **- rh**	
00007642	0001	TimeVariantTypeEnum **- type**	0x03
00007643	0002	array of bytes **- stringValue**	
00007645	002C	**B**: TimeAnimationValueListEntry **- timeAnimValueListEntry[1]**	
00007645	000C	TimeAnimationValueAtom **- timeAnimationValueAtom**	
00007645	0008	RecordHeader **- rh**	
0000764D	0004	signed integer **- time**	0x000003E8
00007651	0015	TimeVariantString **- case of TL_TVT_String**	
00007651	0008	RecordHeader **- rh**	
00007659	0001	TimeVariantTypeEnum **- type**	0x03
0000765A	000C	array of bytes **- stringValue**	ppt_x
00007666	000B	TimeVariantString **- varFormula**	
00007666	0008	RecordHeader **- rh**	
0000766E	0001	TimeVariantTypeEnum **- type**	0x03
0000766F	0002	array of bytes **- stringValue**	

Figure 130: Child-record hierarchy of TimeAnimationValueListContainer

The **TimeAnimationValueListContainer** (section 2.8.31) record specifies two key points of the animation. The TimeAnimationValueListEntry record A specifies the starting point, while the TimeAnimationValueListEntry record B specifies the endpoint.

timeAnimValueListEntry[0].timeAnimationValueAtom.time: 0x00000000 specifies that the first key point is the starting point of the animation.

timeAnimValueListEntry[0].case of TL_TVT_String.stringValue: "ppt_x" specifies the formula that is used to calculate the property value at time 0x00000000. The value of this formula is the original value of the position of the shape on the horizontal axis.

timeAnimValueListEntry[1].timeAnimationValueAtom.time: 0x000003E8 specifies that the second key point is the endpoint of the animation.

timeAnimValueListEntry[1].case of TL_TVT_String.stringValue: "ppt_x" specifies the formula that is used to calculate the property value at time 0x000003E8. The value of this formula is the original value of the position of the shape on the horizontal axis.

Because the values at the two key points are identical and equal to the original value of the position on the horizontal axis, the shape does not change position on the horizontal axis during the sink-down animation.

The child-record hierarchy of the **TimeBehaviorContainer** record B from the third table titled "TimeAnimateBehaviorContainer child-record hierarchy in the sixth-level ExtTimeNodeContainer" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00007671	0061	TimeBehaviorContainer **- behavior**	
00007671	0008	RecordHeader **- rh**	
00007679	0018	**A**: TimeBehaviorAtom **- behaviorAtom**	
00007679	0008	RecordHeader **- rh**	
00007679	1 bit	bit **- fAdditivePropertyUsed**	0x0
00007679	1 bit	bit **- reserved1**	0x0
00007679	1 bit	bit **- fAttributeNamesPropertyUsed**	0x1
00007679	1 bit	bit **- reserved2**	0x0
00007679	28 bits	unsigned integer **- reserved3**	0x0000000
00007685	0004	unsigned integer **- behaviorAdditive**	0x00000000
00007689	0004	signed integer **- behaviorAccumulate**	0x00000000
0000768D	0004	signed integer **- behaviorTransform**	0x00000000
00007691	001D	TimeStringListContainer **- stringList**	
00007691	0008	RecordHeader **- rh**	
00007699	0015	TimeVariantString **- string**	
00007699	0008	RecordHeader **- rh**	
000076A1	0001	TimeVariantTypeEnum **- type**	0x03
000076A2	000C	array of bytes **- stringValue**	ppt_x
000076AE	0024	**B**: ClientVisualElementContainer **- clientVisualElement**	

Figure 131: Child-record hierarchy of TimeBehaviorContainer

The TimeBehaviorAtom record A is similar to the record as specified in the second table titled "TimeBehaviorAtom record A child-record hierarchy" in this section. The

ClientVisualElementContainer (section 2.8.44) record B specifies that the target of the animation is the shape.

stringList.string.stringValue: "ppt_x" specifies that the property to be animated is the position of the shape on the horizontal axis.

The child-record hierarchy of the **ExtTimeNodeContainer** record G from the third table titled "Fifth-level ExtTimeNodeContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
000076D2	012D	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
000076D2	0008	RecordHeader **- rh**	
000076DA	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
00007702	0008	TimePropertyList4TimeNodeContainer **- timePropertyList**	
00007702	0008	RecordHeader **- rh**	
0000770A	00E5	**B**: TimeAnimateBehaviorContainer **- timeAnimateBehavior**	
000077EF	0010	**C**: TimeModifierAtom **- timeModifierAtom**	
000077EF	0008	RecordHeader **- rh**	
000077F7	0004	TimeModifierEnum **- type**	0x00000004
000077FB	0004	unsigned integer **- value**	0x3F800000

Figure 132: The third sixth-level ExtTimeNodeContainer child-record hierarchy

This time node contains a generic behavior that animates the position of the shape on the vertical axis. The TimeModifierAtom record C specifies the deceleration attribute that the animation will use.

timeModifierAtom.type: 0x00000004 specifies that the **timeModifierAtom.value** field defines the deceleration of the generic behavior as specified in the **TimeAnimateBehaviorContainer** (section 2.8.29) record B.

timeModifierAtom.value: 0x3F800000 specifies that the deceleration of the behavior occurs during the entire length of the animation.

The child-record hierarchy of the TimeNodeAtom record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
000076DA	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
000076DA	0008	RecordHeader **- rh**	
000076E2	0004	unsigned integer **- reserved1**	0x00000000
000076E6	0004	unsigned integer **- restart**	0x00000000
000076EA	0004	TimeNodeTypeEnum **- type**	0x00000003
000076EE	0004	unsigned integer **- fill**	0x00000000
000076F2	0004	signed integer **- reserved2**	0x00000000

Offset	Size	Structure	Value
000076F6	0001	unsigned integer **- reserved3**	0x00
000076FA	0004	signed integer **- duration**	0x00000064
000076FE	1 bit	bit **- fFillProperty**	0x1
000076FE	1 bit	bit **- fRestartProperty**	0x0
000076FE	1 bit	bit **- reserved4**	0x0
000076FE	1 bit	bit **- fGroupingTypeProperty**	0x1
000076FE	1 bit	bit **- fDurationProperty**	0x1
000076FE	27 bits	unsigned integer **- reserved5**	0x0000000

Figure 133: TimeNodeAtom record A child-record hierarchy in the sixth-level ExtTimeNodeContainer

duration: 0x00000064 specifies that the duration of the animation is 0.1 seconds.

The child-record hierarchy of the **TimeAnimateBehaviorContainer** (section 2.8.29) record B from the second table titled "The third sixth-level ExtTimeNodeContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
0000770A	00E5	TimeAnimateBehaviorContainer **- timeAnimateBehavior**	
0000770A	0008	RecordHeader **- rh**	
00007712	0014	**A**: TimeAnimateBehaviorAtom **- animateBehaviorAtom**	
00007712	0008	RecordHeader **- rh**	
0000771A	0004	unsigned integer **- calcMode**	0x00000001
0000771E	0004	unsigned integer **- fByPropertyUsed**	0x00000000
0000771E	0004	unsigned integer **- fFromPropertyUsed**	0x00000000
0000771E	0004	unsigned integer **- fToPropertyUsed**	0x00000000
0000771E	0004	unsigned integer **- fCalcModePropertyUsed**	0x00000001
0000771E	0004	unsigned integer **- fAnimationValuesPropertyUsed**	0x00000001
0000771E	0004	unsigned integer **- fValueTypePropertyUsed**	0x00000001
0000771E	0004	unsigned integer **- reserved**	0x00000000
00007722	0004	TimeAnimateBehaviorValueTypeEnum **- valueType**	0x00000001
00007726	0068	**B**: TimeAnimationValueListContainer **- animateValueList**	
0000778E	0061	**C**: TimeBehaviorContainer **- behavior**	

Figure 134: TimeAnimateBehaviorContainer child-record hierarchy in the sixth-level ExtTimeNodeContainer

The TimeAnimateBehaviorAtom record A is similar to the TimeAnimateBehaviorAtom record as specified in the third table titled "TimeAnimateBehaviorContainer child-record hierarchy in the sixth-level ExtTimeNodeContainer" in this section.

The child-record hierarchy of the **TimeAnimationValueListContainer** (section 2.8.31) record B from the fourth table titled "TimeAnimateBehaviorContainer child-record hierarchy in the sixth-level ExtTimeNodeContainer" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00007726	0068	TimeAnimationValueListContainer **- animateValueList**	
00007726	0008	RecordHeader **- rh**	
0000772E	002C	**A**: TimeAnimationValueListEntry **- timeAnimValueListEntry[0]**	
0000772E	000C	TimeAnimationValueAtom **- timeAnimationValueAtom**	
0000772E	0008	RecordHeader **- rh**	
00007736	0004	signed integer **- time**	0x00000000
0000773A	0015	TimeVariantString **- case of TL_TVT_String**	
0000773A	0008	RecordHeader **- rh**	
00007742	0001	TimeVariantTypeEnum **- type**	0x03
00007743	000C	array of bytes **- stringValue**	ppt_y
0000774F	000B	TimeVariantString **- varFormula**	
0000774F	0008	RecordHeader **- rh**	
00007757	0001	TimeVariantTypeEnum **- type**	0x03
00007758	0002	array of bytes **- stringValue**	
0000775A	0034	**B**: TimeAnimationValueListEntry **- timeAnimValueListEntry[1]**	
0000775A	000C	TimeAnimationValueAtom **- timeAnimationValueAtom**	
0000775A	0008	RecordHeader **- rh**	
00007762	0004	signed integer **- time**	0x000003E8
00007766	001D	TimeVariantString **- case of TL_TVT_String**	
00007766	0008	RecordHeader **- rh**	
0000776E	0001	TimeVariantTypeEnum **- type**	0x03
0000776F	0014	array of bytes **- stringValue**	ppt_y-.03
00007783	000B	TimeVariantString **- varFormula**	
00007783	0008	RecordHeader **- rh**	
0000778B	0001	TimeVariantTypeEnum **- type**	0x03
0000778C	0002	array of bytes **- stringValue**	

Figure 135: TimeAnimationValueListContainer child-record hierarchy

timeAnimValueListEntry[0].timeAnimationValueAtom.time: 0x00000000 specifies that the first key point is the starting point of the animation.

timeAnimValueListEntry[0].case of TL_TVT_String.stringValue: "ppt_y" specifies the formula that is used to calculate the property value at time 0x00000000. The value of this formula is the original value of the position of the shape on the vertical axis.

timeAnimValueListEntry[1].timeAnimationValueAtom.time: 0x000003E8 specifies that the second key point is the endpoint of the animation.

timeAnimValueListEntry[1].case of TL_TVT_String.stringValue: "ppt_y-.03" specifies the formula that is used to calculate the property value at time 0x000003E8. The value of this formula moves the shape above its original position by a small amount in the vertical direction.

The child-record hierarchy of the **TimeBehaviorContainer** record C from the fourth table titled "TimeAnimateBehaviorContainer child-record hierarchy in the sixth-level ExtTimeNodeContainer" in s section is shown expanded in the following table.

Offset	Size	Structure	Value
0000778E	0061	TimeBehaviorContainer **- behavior**	
0000778E	0008	RecordHeader **- rh**	
00007796	0018	**A**: TimeBehaviorAtom **- behaviorAtom**	
00007796	0008	RecordHeader **- rh**	
00007796	1 bit	bit **- fAdditivePropertyUsed**	0x0
00007796	1 bit	bit **- reserved1**	0x0
00007796	1 bit	bit **- fAttributeNamesPropertyUsed**	0x1
00007796	1 bit	bit **- reserved2**	0x0
00007796	28 bits	unsigned integer **- reserved3**	0x0000000
000077A2	0004	unsigned integer **- behaviorAdditive**	0x00000000
000077A6	0004	signed integer **- behaviorAccumulate**	0x00000000
000077AA	0004	signed integer **- behaviorTransform**	0x00000000
000077AE	001D	TimeStringListContainer **- stringList**	
000077AE	0008	RecordHeader **- rh**	
000077B6	0015	TimeVariantString **- string**	
000077B6	0008	RecordHeader **- rh**	
000077BE	0001	TimeVariantTypeEnum **- type**	0x03
000077BF	000C	array of bytes **- stringValue**	ppt_y
000077CB	0024	**B**: ClientVisualElementContainer **- clientVisualElement**	

Figure 23: TimeBehaviorContainer child-record hierarchy

The TimeBehaviorAtom record A is similar to the TimeBehaviorAtom record as specified in the second table titled "TimeBehaviorAtom record A child-record hierarchy" in this section. The **ClientVisualElementContainer** record B specifies that the target of the animation is the shape.

stringList.string.stringValue: "ppt_y" specifies that the property to be animated is the position of the shape on the vertical axis.

The child-record hierarchy of the **ExtTimeNodeContainer** record F from the third table titled "Fifth-level ExtTimeNodeContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
000077FF	0149	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
000077FF	0008	RecordHeader **- rh**	
00007807	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
0000782F	0008	**B**: TimePropertyList4TimeNodeContainer **- timePropertyList**	
00007837	00E1	**C**: TimeAnimateBehaviorContainer **- timeAnimateBehavior**	
00007918	0020	**D**: TimeConditionContainer **- timeCondition**	
00007918	0008	RecordHeader **- rh**	
00007920	0018	TimeConditionAtom **- conditionAtom**	
00007920	0008	RecordHeader **- rh**	
00007928	0004	TriggerObjectEnum **- triggerObject**	0x00000000
0000792C	0004	unsigned integer **- triggerEvent**	0x00000000
00007930	0004	unsigned integer **- id**	0x00000000
00007934	0004	signed integer **- delay**	0x00000064
00007938	0010	**E**: TimeModifierAtom **- timeModifierAtom**	
00007938	0008	RecordHeader **- rh**	
00007940	0004	TimeModifierEnum **- type**	0x00000003
00007944	0004	unsigned integer **- value**	0x3F800000

Figure 137: The fourth sixth-level ExtTimeNodeContainer child-record hierarchy

This time node contains a generic behavior that animates the position of the shape on the vertical axis. The **TimeConditionContainer** record D specifies that this behavior is deferred until the behaviors contained in the **ExtTimeNodeContainer** records F and G from the third table titled "Fifth-level ExtTimeNodeContainer child-record hierarchy" in this section start. The **TimePropertyList4TimeNodeContainer** record B is similar to the **TimePropertyList4TimeNodeContainer** record as specified in the second table titled "The third sixth-level ExtTimeNodeContainer child-record hierarchy" in this section.

timeCondition.conditionAtom.delay: 0x00000064 specifies that this time node is delayed 0.1 seconds after its parent time node at the fifth level is activated.

timeModifierAtom.type: 0x00000003 specifies that the **timeModifierAtom.value** field defines the acceleration of the rotation animation.

timeModifierAtom.value: 0x3F800000 specifies the acceleration occurs over the entire duration of the behavior.

The child-record hierarchy of the TimeNodeAtom record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00007807	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
00007807	0008	RecordHeader **- rh**	
0000780F	0004	unsigned integer **- reserved1**	0x00000000
00007813	0004	unsigned integer **- restart**	0x00000000
00007817	0004	TimeNodeTypeEnum **- type**	0x00000003
0000781B	0004	unsigned integer **- fill**	0x00000000
0000781F	0004	signed integer **- reserved2**	0x00000000
00007823	0001	unsigned integer **- reserved3**	0x00
00007827	0004	signed integer **- duration**	0x00000384
0000782B	1 bit	bit **- fFillProperty**	0x1
0000782B	1 bit	bit **- fRestartProperty**	0x0
0000782B	1 bit	bit **- reserved4**	0x0
0000782B	1 bit	bit **- fGroupingTypeProperty**	0x1
0000782B	1 bit	bit **- fDurationProperty**	0x1
0000782B	27 bits	unsigned integer **- reserved5**	0x0000000

Figure 138: TimeNodeAtom record A child-record hierarchy in the sixth-level ExtTimeNodeContainer

duration: 0x00000384 specifies that the duration of this time node is 0.9 seconds.

The child-record hierarchy of the **TimeAnimateBehaviorContainer** (section 2.8.29) record C from the table titled "The fourth sixth-level ExtTimeNodeContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00007837	00E1	TimeAnimateBehaviorContainer **- timeAnimateBehavior**	
00007837	0008	RecordHeader **- rh**	
0000783F	0014	**A**: TimeAnimateBehaviorAtom **- animateBehaviorAtom**	
0000783F	0008	RecordHeader **- rh**	
00007847	0004	unsigned integer **- calcMode**	0x00000001
0000784B	1 bit	bit **- fByPropertyUsed**	0x0
0000784B	1 bit	bit **- fFromPropertyUsed**	0x0
0000784B	1 bit	bit **- fToPropertyUsed**	0x0
0000784B	1 bit	bit **- fCalcModePropertyUsed**	0x1

Offset	Size	Structure	Value
0000784B	1 bit	bit **- fAnimationValuesPropertyUsed**	0x1
0000784B	1 bit	bit **- fValueTypePropertyUsed**	0x1
0000784B	26 bits	unsigned integer **- reserved**	0x0000000
0000784F	0004	TimeAnimateBehaviorValueTypeEnum **- valueType**	0x00000001
00007853	0064	**B**: TimeAnimationValueListContainer **- animateValueList**	
000078B7	0061	**C**: TimeBehaviorContainer **- behavior**	

Figure 139: TimeAnimateBehaviorContainer child-record hierarchy in the sixth-level ExtTimeNodeContainer

The TimeAnimateBehaviorAtom record A is similar to the TimeAnimateBehaviorAtom record as specified in the third table titled "TimeAnimateBehaviorContainer child-record hierarchy in the sixth-level ExtTimeNodeContainer" in this section.

The child-record hierarchy of the **TimeAnimationValueListContainer** (section 2.8.31) record B from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00007853	0064	TimeAnimationValueListContainer **- animateValueList**	
00007853	0008	RecordHeader **- rh**	
0000785B	002C	**A**: TimeAnimationValueListEntry **- timeAnimValueListEntry[0]**	
0000785B	000C	TimeAnimationValueAtom **- timeAnimationValueAtom**	
0000785B	0008	RecordHeader **- rh**	
00007863	0004	signed integer **- time**	0x00000000
00007867	0015	TimeVariantString **- case of TL_TVT_String**	
00007867	0008	RecordHeader **- rh**	
0000786F	0001	TimeVariantTypeEnum **- type**	0x03
00007870	000C	array of bytes **- stringValue**	ppt_y
0000787C	000B	TimeVariantString **- varFormula**	
0000787C	0008	RecordHeader **- rh**	
00007884	0001	TimeVariantTypeEnum **- type**	0x03
00007885	0002	array of bytes **- stringValue**	
00007887	0030	**B**: TimeAnimationValueListEntry **- timeAnimValueListEntry[1]**	
00007887	000C	TimeAnimationValueAtom **- timeAnimationValueAtom**	
00007887	0008	RecordHeader **- rh**	
0000788F	0004	signed integer **- time**	0x000003E8
00007893	0019	TimeVariantString **- case of TL_TVT_String**	

Offset	Size	Structure	Value
00007893	0008	RecordHeader **- rh**	
0000789B	0001	TimeVariantTypeEnum **- type**	0x03
0000789C	0010	array of bytes **- stringValue**	ppt_y+1
000078AC	000B	TimeVariantString **- varFormula**	
000078AC	0008	RecordHeader **- rh**	
000078B4	0001	TimeVariantTypeEnum **- type**	0x03
000078B5	0002	array of bytes **- stringValue**	

Figure 140: TimeAnimationValueListContainer child-record hierarchy

The **TimeAnimationValueListContainer** record (section 2.8.31) specifies two key points of the animation. The TimeAnimationValueListEntry record A specifies the starting point, while the TimeAnimationValueListEntry record B specifies the endpoint.

timeAnimValueListEntry[0].timeAnimationValueAtom.time: 0x00000000 specifies that the first key point is the starting point of the animation.

timeAnimValueListEntry[0].case of TL_TVT_String.stringValue: "ppt_y" specifies the formula that is used to calculate the property value at time 0x00000000. The value of this formula is the original value of the position of the shape on the vertical axis.

timeAnimValueListEntry[1].timeAnimationValueAtom.time: 0x000003E8 specifies that the second key point is the endpoint of the animation.

timeAnimValueListEntry[1].case of TL_TVT_String.stringValue: "ppt_y+1" specifies the formula that is used to calculate the property value at time 0x000003E8. The value of this formula makes sure that the shape is moved beneath the bottom of slide show where the shape is invisible.

This behavior moves the shape from its original position to a position beneath the slide show so that the shape is invisible.

The child-record hierarchy of the **TimeBehaviorContainer** record C from the fifth table titled "TimeAnimateBehaviorContainer child-record hierarchy in the sixth-level ExtTimeNodeContainer" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
000078B7	0061	TimeBehaviorContainer **- behavior**	
000078B7	0008	RecordHeader **- rh**	
000078BF	0018	**A**: TimeBehaviorAtom **- behaviorAtom**	
000078BF	0008	RecordHeader **- rh**	
000078BF	0004	unsigned integer **- fAdditivePropertyUsed**	0x00000000
000078BF	0004	unsigned integer **- reserved1**	0x00000000
000078BF	0004	unsigned integer **- fAttributeNamesPropertyUsed**	0x00000001
000078BF	0004	unsigned integer **- reserved2**	0x00000000
000078BF	0004	unsigned integer **- reserved3**	0x00000000

Offset	Size	Structure	Value
000078CB	0004	unsigned integer **- behaviorAdditive**	0x00000000
000078CF	0004	signed integer **- behaviorAccumulate**	0x00000000
000078D3	0004	signed integer **- behaviorTransform**	0x00000000
000078D7	001D	TimeStringListContainer **- stringList**	
000078D7	0008	RecordHeader **- rh**	
000078DF	0015	TimeVariantString **- string**	
000078DF	0008	RecordHeader **- rh**	
000078E7	0001	TimeVariantTypeEnum **- type**	0x03
000078E8	000C	array of bytes **- stringValue**	ppt_y
000078F4	0024	**B**: ClientVisualElementContainer **- clientVisualElement**	

Figure 141: Child-record hierarchy of TimeBehaviorContainer

The TimeBehaviorAtom record A is similar to the TimeBehaviorAtom record as specified in the second table titled "TimeBehaviorAtom record A child-record hierarchy" in this section. The **ClientVisualElementContainer** record B specifies that the target of the animation is the shape.

stringList.string.stringValue: "ppt_y" specifies that the property to be animated is the position of the shape on the vertical axis.

The child-record hierarchy of the **ExtTimeNodeContainer** record G from the third table titled "Fifth-level ExtTimeNodeContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00007948	00FE	ExtTimeNodeContainer **- case of RT_TimeExtTimeNodeContainer**	
00007948	0008	RecordHeader **- rh**	
00007950	0028	**A**: TimeNodeAtom **- timeNodeAtom**	
00007978	0008	**B**: TimePropertyList4TimeNodeContainer **- timePropertyList**	
00007980	00A6	**C**: TimeSetBehaviorContainer **- timeSetBehavior**	
00007A26	0020	**D**: TimeConditionContainer **- timeCondition**	
00007A26	0008	RecordHeader **- rh**	
00007A2E	0018	TimeConditionAtom **- conditionAtom**	
00007A2E	0008	RecordHeader **- rh**	
00007A36	0004	TriggerObjectEnum **- triggerObject**	0x00000000
00007A3A	0004	unsigned integer **- triggerEvent**	0x00000000
00007A3E	0004	unsigned integer **- id**	0x00000000
00007A42	0004	signed integer **- delay**	0x000003E7

Figure 142: The fifth sixth-level ExtTimeNodeContainer child-record hierarchy

This time node contains a set behavior that hides the shape. The TimeNodeAtom record A is similar to the record specified in the second table titled "TimeNodeAtom record A child-record hierarchy in the sixth-level ExtTimeNodeContainer" in this section. The **TimePropertyList4TimeNodeContainer** record B is similar to the **TimePropertyList4TimeNodeContainer** record as specified in the second table titled "The second sixth-level ExtTimeNodeContainer child-record hierarchy" in this section. The **TimeConditionContainer** record D specifies that this behavior is deferred until the last millisecond.

timeCondition.conditionAtom.delay: 0x000003E7 specifies that this time node will start 0.999 seconds after the parent time node at the fifth level is activated.

The child-record hierarchy of the **TimeSetBehaviorContainer** record C from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00007980	00A6	TimeSetBehaviorContainer **- timeSetBehavior**	
00007980	0008	RecordHeader **- rh**	
00007988	0010	**A**: TimeSetBehaviorAtom **- setBehaviorAtom**	
00007988	0008	RecordHeader **- rh**	
00007988	0004	unsigned integer **- fToPropertyUsed**	0x00000001
00007988	0004	unsigned integer **- fValueTypePropertyUsed**	0x00000000
00007988	0004	unsigned integer **- reserved**	0x00000000
00007994	0004	TimeAnimateBehaviorValueTypeEnum **- valueType**	0x00000001
00007998	0017	TimeVariantString **- varTo**	
00007998	0008	RecordHeader **- rh**	
000079A0	0001	TimeVariantTypeEnum **- type**	0x03
000079A1	000E	array of bytes **- stringValue**	hidden
000079AF	0077	**B**: TimeBehaviorContainer **- behavior**	

Figure 143: TimeSetBehaviorContainer child-record hierarchy in the sixth-level ExtTimeNodeContainer

The TimeSetBehaviorAtom record A is similar to the TimeSetBehaviorAtom record as specified in the table titled "Child-record hierarchy of TimeSetBehaviorContainer in the sixth-level ExtTimeNodeContainer" in section 3.7.1.

varTo.stringValue: "hidden" specifies that the shape is invisible.

The child-record hierarchy of the **TimeBehaviorContainer** record B from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
000079AF	0077	TimeBehaviorContainer **- behavior**	
000079AF	0008	RecordHeader **- rh**	
000079B7	0018	**A**: TimeBehaviorAtom **- behaviorAtom**	
000079B7	0008	RecordHeader **- rh**	

Offset	Size	Structure	Value
000079B7	1 bit	bit **- fAdditivePropertyUsed**	0x0
000079B7	1 bit	bit **- reserved1**	0x0
000079B7	1 bit	bit **- fAttributeNamesPropertyUsed**	0x1
000079B7	1 bit	bit **- reserved2**	0x0
000079B7	28 bits	unsigned integer **- reserved3**	0x0000000
000079C3	0004	unsigned integer **- behaviorAdditive**	0x00000000
000079C7	0004	signed integer **- behaviorAccumulate**	0x00000000
000079CB	0004	signed integer **- behaviorTransform**	0x00000000
000079CF	0033	TimeStringListContainer **- stringList**	
000079CF	0008	RecordHeader **- rh**	
000079D7	002B	TimeVariantString **- string**	
000079D7	0008	RecordHeader **- rh**	
000079DF	0001	TimeVariantTypeEnum **- type**	0x03
000079E0	0022	array of bytes **- stringValue**	style.visibility
00007A02	0024	**B**: ClientVisualElementContainer **- clientVisualElement**	

Figure 144: TimeBehaviorContainer child-record hierarchy

The TimeBehaviorAtom record A is similar to the TimeBehaviorAtom record as specified in the second table titled "TimeBehaviorAtom record A child-record hierarchy" in this section. The **ClientVisualElementContainer** record B specifies that the target of the animation is the shape.

stringList.string.stringValue: "style.visibility" specifies the property to be animated that controls the visibility of the shape.

3.8 Shape Client Data Example

The following sections provide examples of a shape anchor, a placeholder shape, a shape text body, an OLE object, and an external video.

3.8.1 Shape Anchor Example

This example shows how to locate the shape anchor for the title placeholder shape on presentation slide 6 as shown in figure titled "Presentation slide 6" in section 3.1.

The child-record hierarchy of the **DrawingContainer** (section 2.5.13) record A from the table titled "SlideContainer record U child-record hierarchy" in section 3.5.2 is shown expanded in the following table.

Offset	Size	Structure
00007B58	0423	DrawingContainer **- drawing**
00007B58	0008	RecordHeader **- rh**

Offset	Size	Structure
00007B60	041B	OfficeArtDgContainer **- OfficeArtDg**
00007B60	0008	OfficeArtRecordHeader **- rh**
00007B68	0010	OfficeArtFDG **- drawingData**
00007B78	03B3	OfficeArtSpgrContainer **- groupShape**
00007B78	0008	OfficeArtRecordHeader **- rh**
00007B80	0030	OfficeArtSpContainer **- case of msofbtSpContainer**
00007BB0	0080	**A**: OfficeArtSpContainer **- case of msofbtSpContainer**
00007C30	0109	**B**: OfficeArtSpContainer **- case of msofbtSpContainer**
00007D39	00F7	**C**: OfficeArtSpContainer **- case of msofbtSpContainer**
00007E30	00FB	**D**: OfficeArtSpContainer **- case of msofbtSpContainer**
00007F2B	0050	OfficeArtSpContainer **- shape**

Figure 145: DrawingContainer child-record hierarchy

The child-record hierarchy of the **OfficeArtSpContainer** ([MS-ODRAW] section 2.2.14) record A from the previous table is shown expanded in the following table.

Offset	Size	Structure
00007BB0	0080	OfficeArtSpContainer **- case of msofbtSpContainer**
00007BB0	0008	OfficeArtRecordHeader **- rh**
00007BB8	0010	OfficeArtFSP **- shapeProp**
00007BC8	002C	OfficeArtFOPT **- shapePrimaryOptions**
00007BF4	0010	**A**: OfficeArtClientAnchor **- clientAnchor**
00007C04	0018	**B**: OfficeArtClientData **- clientData**
00007C1C	0014	**C**: OfficeArtClientTextbox **- clientTextbox**

Figure 146: OfficeArtSpContainer child-record hierarchy

The child-record hierarchy of the OfficeArtClientAnchor record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00007BF4	0010	**A**: OfficeArtClientAnchor **- clientAnchor**	
00007BF4	0008	OfficeArtRecordHeader **- rh**	
00007BF4	4 bits	unsigned integer **- recVer**	0x0
00007BF4	12 bits	unsigned integer **- recInstance**	0x000
00007BF6	0002	unsigned integer **- recType**	0xF010
00007BF8	0004	unsigned integer **- recLen**	0x00000008

Offset	Size	Structure	Value
00007BFC	0008	OfficeArtClientAnchorData **- clientAnchorData**	
00007BFC	0008	SmallRectStruct **- case of 0x00000008**	
00007BFC	0002	signed integer **- top**	0x00AD
00007BFE	0002	signed integer **- left**	0x0120
00007C00	0002	signed integer **- right**	0x1560
00007C02	0002	signed integer **- bottom**	0x037D

Figure 147: OfficeArtClientAnchor record A child-record hierarchy

rh.recLen: 0x00000008 specifies that a SmallRectStruct record is contained in **clientAnchorData**.

clientAnchorData.case of 0x00000008.top: 0x00AD specifies the minimum *y*-value of the rectangle.

clientAnchorData.case of 0x00000008.left: 0x0120 specifies the minimum *x*-value of the rectangle.

clientAnchorData.case of 0x00000008.right: 0x1560 specifies the maximum *x*-value of the rectangle.

clientAnchorData.case of 0x00000008.bottom: 0x037D specifies the maximum *y*-value of the rectangle.

3.8.2 Shape Placeholder Example

This example shows how to locate the title placeholder shape on presentation slide 6 as shown in figure titled "Presentation slide 6" in section 3.1.

The child-record hierarchy of the OfficeArtClientData record B from the second table titled "OfficeArtSpContainer child-record hierarchy" in section 3.8.1 is shown expanded in the following table.

Offset	Size	Structure	Value
00007C04	0018	**B**: OfficeArtClientData **- clientData**	
00007C04	0008	OfficeArtRecordHeader **- rh**	
00007C0C	0010	PlaceholderAtom **- placeholderAtom**	
00007C0C	0008	RecordHeader **- rh**	
00007C14	0004	signed integer **- position**	0x00000000
00007C18	0001	PlaceholderEnum **- placementId**	0x0D
00007C19	0001	PlaceholderSize **- size**	0x00

Figure 148: OfficeArtClientData record B child-record hierarchy

placeholderAtom.position: 0x00000000 specifies the identifier for the placeholder shape.

placeholderAtom.placementId: 0x0D specifies that this placeholder shape is a title text placeholder shape.

The child-record hierarchy of the OfficeArtClientTextbox record C from the second table titled "OfficeArtSpContainer child-record hierarchy" in section 3.8.1 is shown expanded in the following table.

Offset	Size	Structure	Value
00007C1C	0014	**C**: OfficeArtClientTextbox **- clientTextbox**	
00007C1C	0008	OfficeArtRecordHeader **- rh**	
00007C24	000C	OutlineTextRefAtom **- case of RT_OutlineTextRefAtom**	
00007C24	0008	RecordHeader **- rh**	
00007C24	4 bits	unsigned integer **- recVer**	0x0
00007C24	12 bits	unsigned integer **- recInstance**	0x000
00007C26	0002	RecordType **- recType**	0x0F9E
00007C28	0004	unsigned integer **- recLen**	0x00000004
00007C2C	0004	signed integer **- index**	0x00000000

Figure 149: OfficeArtClientTextbox child-record hierarchy

case of RT_OutlineTextRefAtom: specifies a reference to text contained in the **SlideListWithTextContainer** record (section 2.4.14.3).

case of RT_OutlineTextRefAtom.index: 0x00000000 specifies the index into the sequence of the TextHeaderAtom records that follows the slide persist record. This specifies record O in the table titled "SlideListWithTextContainer child-record hierarchy" in section 3.4. The table titled "Outline TextBytesAtom record O" in section 3.4 shows record O expanded. The contained text "shapes with text" is specified by record P in the table titled "SlideListWithTextContainer child-record hierarchy" in section 3.4, and is shown expanded in the table titled "Outline TextBytesAtom record P" in section 3.4.

3.8.3 Shape Text Example

This example shows how to locate the text of the triangle shape on presentation slide 6 as shown in figure titled "Presentation slide 6" in section 3.1.

The child-record hierarchy of the **OfficeArtSpContainer** ([MS-ODRAW] section 2.2.14) record C from the table titled "DrawingContainer child-record hierarchy" in section 3.8.1 is shown expanded in the following table.

Offset	Size	Structure
00007D39	00F7	OfficeArtSpContainer **- case of msofbtSpContainer**
00007D39	0008	OfficeArtRecordHeader **- rh**
00007D41	0010	OfficeArtFSP **- shapeProp**
00007D51	0044	OfficeArtFOPT **- shapePrimaryOptions**
00007D95	000E	OfficeArtTertiaryFOPT **- shapeTertiaryOptons**
00007DA3	0010	OfficeArtClientAnchor **- clientAnchor**
00007DB3	007D	**A**: OfficeArtClientTextbox **- clientTextbox**

Figure 24: OfficeArtSpContainer child-record hierarchy

The child-record hierarchy of the OfficeArtClientTextbox record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00007DB3	007D	OfficeArtClientTextbox **- clientTextbox**	
00007DB3	0008	OfficeArtRecordHeader **- rh**	
00007DB3	4 bits	unsigned integer **- recVer**	0xF
00007DB3	12 bits	unsigned integer **- recInstance**	0x000
00007DB5	0002	unsigned integer **- recType**	0xF00D
00007DB7	0004	unsigned integer **- recLen**	0x00000075
00007DBB	000C	**A**: TextHeaderAtom **- case of RT_TextHeaderAtom**	
00007DC7	0019	**B**: TextBytesAtom **- case of RT_TextBytesAtom**	
00007DE0	0050	StyleTextPropAtom **- case of RT_StyleTextPropAtom**	

Figure 25: OfficeArtClientTextbox child-record hierarchy

The child-record hierarchy of the TextHeaderAtom record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00007DBB	000C	**A**: TextHeaderAtom **- case of RT_TextHeaderAtom**	
00007DBB	0008	RecordHeader **- rh**	
00007DBB	4 bits	unsigned integer **- recVer**	0x0
00007DBB	12 bits	unsigned integer **- recInstance**	0x000
00007DBD	0002	RecordType **- recType**	0x0F9F
00007DBF	0004	unsigned integer **- recLen**	0x00000004
00007DC3	0004	TextTypeEnum **- textType**	0x00000004

Figure 26: TextHeaderAtom record A child-record hierarchy

textType: 0x00000004 specifies that the body of text is type Tx_TYPE_OTHER.

The child-record hierarchy of the TextBytesAtom record B from the table titled "OfficeArtClientTextbox child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00007DC7	0019	**B**: TextBytesAtom **- case of RT_TextBytesAtom**	
00007DC7	0008	RecordHeader **- rh**	
00007DC7	4 bits	unsigned integer **- recVer**	0x0
00007DC7	12 bits	unsigned integer **- recInstance**	0x000

Offset	Size	Structure	Value
00007DC9	0002	RecordType **- recType**	0x0FA8
00007DCB	0004	unsigned integer **- recLen**	0x00000011
00007DCF	0011	array of bytes **- textBytes**	a \vbold\v triangle

Figure 27: TextBytesAtom record B child-record hierarchy

textBytes: "a \vbold\v triangle" specifies the characters of the text body.

3.8.4 OLE Object Example

This example shows how to locate the pie chart on presentation slide 5 as shown in figure titled "Presentation slide 5" in section 3.1.

The child-record hierarchy of the **DrawingContainer** (section 2.5.13) record A from the table titled "SlideContainer record J child-record hierarchy" in section 3.5.2 is shown expanded in the following table.

Offset	Size	Structure
00003EAB	01B8	DrawingContainer **- drawing**
00003EAB	0008	RecordHeader **- rh**
00003EB3	01B0	OfficeArtDgContainer **- OfficeArtDg**
00003EB3	0008	OfficeArtRecordHeader **- rh**
00003EBB	0010	OfficeArtFDG **- drawingData**
00003ECB	0148	OfficeArtSpgrContainer **- groupShape**
00003ECB	0008	OfficeArtRecordHeader **- rh**
00003ED3	0030	OfficeArtSpContainer **- case of msofbtSpContainer**
00003F03	007A	OfficeArtSpContainer **- case of msofbtSpContainer**
00003F7D	0096	**A**: OfficeArtSpContainer **- case of msofbtSpContainer**
00004013	0050	OfficeArtSpContainer **- shape**

Figure 28: OfficeArtSpContainer child-record hierarchy

The child-record hierarchy of the **OfficeArtSpContainer** ([MS-ODRAW] section 2.2.14) record A from the previous table is shown expanded in the following table.

Offset	Size	Structure
00003F7D	0096	OfficeArtSpContainer **- case of msofbtSpContainer**
00003F7D	0008	OfficeArtRecordHeader **- rh**
00003F85	0010	OfficeArtFSP **- shapeProp**
00003F95	004A	OfficeArtFOPT **- shapePrimaryOptions**
00003FDF	0010	OfficeArtClientAnchor **- clientAnchor**

Offset	Size	Structure
00003FEF	0024	**A**: OfficeArtClientData **- clientData**

Figure 29: OfficeArtSpContainer child-record hierarchy

The child-record hierarchy of the **OfficeArtClientData** (section 2.7.3) record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00003FEF	0024	**A**: OfficeArtClientData **- clientData**	
00003FEF	0008	OfficeArtRecordHeader **- rh**	
00003FF7	000C	ExObjRefAtom **- exObjRefAtom**	
00003FF7	0008	RecordHeader **- rh**	
00003FFF	0004	unsigned integer **- exObjIdRef**	0x00000014
00004003	0010	PlaceholderAtom **- placeholderAtom**	

Figure 30: OfficeArtClientData record A child-record hierarchy

exObjRefAtom.exObjIdRef: 0x00000014 specifies the identifier reference to the external object in the **ExObjListContainer** record (section 2.10.1) at the document level.

The child-record hierarchy of the **ExObjListContainer** record F from the table titled "DocumentContainer child-record hierarchy" in section 3.4 is shown expanded in the following table.

Offset	Size	Structure	Value
00005CA1	0152	ExObjListContainer **- exObjList**	
00005CA1	0008	RecordHeader **- rh**	
00005CA9	000C	ExObjListAtom **- exObjListAtom**	
00005CB5	00A2	ExOleEmbedContainer **- case of RT_ExternalOleEmbed**	
00005CB5	0008	RecordHeader **- rh**	
00005CB5	4 bits	unsigned integer **- recVer**	0xF
00005CB5	12 bits	unsigned integer **- recInstance**	0x000
00005CB7	0002	RecordType **- recType**	0x0FCC
00005CB9	0004	unsigned integer **- recLen**	0x0000009A
00005CBD	0010	**A**: ExOleEmbedAtom **- exOleEmbedAtom**	
00005CCD	0020	**B**: ExOleObjAtom **- exOleObjAtom**	
00005CED	0012	**C**: MenuNameAtom **- menuNameAtom**	
00005CFF	0026	**D**: ProgIDAtom **- progIdAtom**	
00005D25	0032	**E**: ClipboardNameAtom **- clipboardNameAtom**	
00005D57	003E	**F**: ExMCIMovieContainer **- case of RT_ExternalMciMovie**	

Offset	Size	Structure	Value
00005D95	005E	**H**: ExHyperlinkContainer **- case of RT_ExternalHyperlink**	

Figure 31: ExObjListContainer child-record hierarchy

The child-record hierarchy of the ExOleEmbedAtom record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00005CBD	0010	**A**: ExOleEmbedAtom **- exOleEmbedAtom**	
00005CBD	0008	RecordHeader **- rh**	
00005CC5	0004	ExColorFollowEnum **- exColorFollow**	0x00000001
00005CC9	0001	unsigned integer **- fCantLockServer**	0x00
00005CCA	0001	unsigned integer **- fNoSizeToServer**	0x00
00005CCB	0001	unsigned integer **- fIsTable**	0x00

Figure 32: ExOleEmbedAtom record A child-record hierarchy

exColorFollow: 0x00000001 specifies that the OLE object follows the color scheme.

fCantLockServer: 0x00 specifies that the OLE server can be locked.

fNoSizeToServer: 0x00 specifies that sending dimensions to the OLE server is required.

fIsTable: 0x00 specifies that the OLE object is not a table.

The child-record hierarchy of the **ExOleObjAtom** (section 2.10.12) record B from the table titled "ExObjListContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00005CCD	0020	**B**: ExOleObjAtom **- exOleObjAtom**	
00005CCD	0008	RecordHeader **- rh**	
00005CD5	0004	DataViewAspectEnum **- drawAspect**	0x00000001
00005CD9	0004	ExOleObjTypeEnum **- type**	0x00000000
00005CDD	0004	unsigned integer **- exObjId**	0x00000014
00005CE1	0004	ExOleObjSubTypeEnum **- subType**	0x00000004
00005CE5	0004	PersistIdRef **- persistIdRef**	0x00000007
00005CE9	0004	BOOL **- unused**	0x00139600

Figure 33: ExOleObjAtom record B child-record hierarchy

drawAspect: 0x00000001 specifies that it is preferable to display the OLE object as an embedded object inside of the container document.

type: 0x00000000 specifies that this OLE object is serialized and saved with the document.

exObjId: 0x00000014 specifies the identifier of this OLE object.

subType: 0x00000004 specifies that the OLE object is created by ProgID "MSGraph.Chart" or "MSGraph".

persistIdRef: 0x00000007 specifies the value to look up in the persist object directory, shown in the second table in section 3.2, to find the persist object stream offset 0x00004D69. This offset matches the offset for the **ExOleObjStg** (section 2.10.34) record P in the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3. This **ExOleObjStg** record represents the OLE object as shown in figure titled "Presentation slide 5" in section 3.1.

The child-record hierarchy of the **ExOleObjStg** record P from the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3 is shown expanded in the following table.

Offset	Size	Structure	Value
00004D69	0E88	**P**: ExOleObjStg	
00004D69	0E88	ExOleObjStgCompressedAtom **- case of 1**	
00004D69	0008	RecordHeader **- rh**	
00004D69	4 bits	unsigned integer **- recVer**	0x0
00004D69	12 bits	unsigned integer **- recInstance**	0x001
00004D6B	0002	RecordType **- recType**	0x1011
00004D6D	0004	unsigned integer **- recLen**	0x00000E80
00004D71	0004	unsigned integer **- decompressedSize**	0x00006A00
00004D75	0E7C	array of bytes **- oleStgCompressed**	78 9C ED 5C 7B 70 94 D5 15 ...

Figure 34: ExOleObjStg record P child-record hierarchy

case of 1.rh.recInstance: 0x001 specifies that this is an ExOleObjStgCompressedAtom record.

case of 1.decompressedSize: 0x00006A00 specifies that the storage length is 0x00006A00 bytes after decompression.

case of 1.oleStgCompressed: 78 9C ED 5C 7B 70 94 D5 15 ... specifies the array data of the compressed OLE object.

The child-record hierarchy of the MenuNameAtom record C from the table titled "ExObjListContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00005CED	0012	**C**: MenuNameAtom **- menuNameAtom**	
00005CED	0008	RecordHeader **- rh**	
00005CED	4 bits	unsigned integer **- recVer**	0x0
00005CED	12 bits	unsigned integer **- recInstance**	0x001
00005CEF	0002	RecordType **- recType**	0x0FBA
00005CF1	0004	unsigned integer **- recLen**	0x0000000A
00005CF5	000A	PrintableUnicodeString **- menuName**	Chart

Figure 35: MenuNameAtom record C child-record hierarchy

menuName: "Chart" specifies the short name of the OLE object.

The child-record hierarchy of the ProgIDAtom record D from the table titled "ExObjListContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00005CFF	0026	**D**: ProgIDAtom **- progIdAtom**	
00005CFF	0008	RecordHeader **- rh**	
00005CFF	4 bits	unsigned integer **- recVer**	0x0
00005CFF	12 bits	unsigned integer **- recInstance**	0x002
00005D01	0002	RecordType **- recType**	0x0FBA
00005D03	0004	unsigned integer **- recLen**	0x0000001E
00005D07	001E	PrintableUnicodeString **- progId**	MSGraph.Chart.8

Figure 36: ProgIDAtom record D child-record hierarchy

progId: "MSGraph.Chart.8" specifies the **ProgID** of the OLE object.

The child-record hierarchy of the ClipboardNameAtom record E from the table titled "ExObjListContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00005D25	0032	**E**: ClipboardNameAtom **- clipboardNameAtom**	
00005D25	0008	RecordHeader **- rh**	
00005D25	4 bits	unsigned integer **- recVer**	0x0
00005D25	12 bits	unsigned integer **- recInstance**	0x003
00005D27	0002	RecordType **- recType**	0x0FBA
00005D29	0004	unsigned integer **- recLen**	0x0000002A
00005D2D	002A	PrintableUnicodeString **- clipboardName**	Microsoft Graph Chart

Figure 37: ClipboardNameAtom record E child-record hierarchy

clipboardName: "Microsoft Graph Chart" specifies the long name of the OLE object.

3.8.5 External Video Example

This example shows how to locate the external video object on presentation slide 4 as shown in figure titled "Presentation slide 4" in section 3.1.

The child-record hierarchy of the **DrawingContainer** (section 2.5.13) record A from the table titled "SlideContainer record I child-record hierarchy" in section 3.5.2 is shown expanded in the following table.

Offset	Size	Structure
000037DD	01BA	DrawingContainer **- drawing**

Offset	Size	Structure
000037DD	0008	RecordHeader **- rh**
000037E5	01B2	OfficeArtDgContainer **- OfficeArtDg**
000037E5	0008	OfficeArtRecordHeader **- rh**
000037ED	0010	OfficeArtFDG **- drawingData**
000037FD	014A	OfficeArtSpgrContainer **- groupShape**
000037FD	0008	OfficeArtRecordHeader **- rh**
00003805	0030	OfficeArtSpContainer **- case of msofbtSpContainer**
00003835	007A	**A**: OfficeArtSpContainer **- case of msofbtSpContainer**
000038AF	0098	OfficeArtSpContainer **- case of msofbtSpContainer**
00003947	0050	OfficeArtSpContainer **- shape**

Figure 38: DrawingContainer child-record hierarchy

The child-record hierarchy of the OfficeArtSpContainer ([MS-ODRAW] section 2.2.14) record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
000038AF	0098	**A**: OfficeArtSpContainer **- case of msofbtSpContainer**	
000038AF	0008	OfficeArtRecordHeader **- rh**	
000038B7	0010	OfficeArtFSP **- shapeProp**	
000038C7	002C	OfficeArtFOPT **- shapePrimaryOptions**	
000038F3	0010	OfficeArtClientAnchor **- clientAnchor**	
00003903	0044	OfficeArtClientData **- clientData**	
00003903	0008	OfficeArtRecordHeader **- rh**	
0000390B	000C	ExObjRefAtom **- exObjRefAtom**	
0000390B	0008	RecordHeader **- rh**	
00003913	0004	unsigned integer **- exObjIdRef**	0x0000001B
00003917	0020	MouseClickInteractiveInfoContainer **- mouseClickInteractiveInfo**	
00003937	0010	PlaceholderAtom **- placeholderAtom**	

Figure 39: OfficeArtSpContainer record A child-record hierarchy

clientData.exObjRefAtom.exObjIdRef: 0x0000001B specifies a reference to an external object with **exObjId** value 0x0000001B in the **ExObjListContainer** record (section 2.10.1) at the document level.

The **ExObjListContainer** record is shown as record F in the table titled "DocumentContainer child-record hierarchy" in section 3.4, and is shown expanded in the table titled "ExObjListContainer child-record hierarchy" in this section. The child-record hierarchy of ExMCIMovieContainer record H is shown expanded in the following table.

Offset	Size	Structure	Value
00005D57	003E	**H**: ExMCIMovieContainer **- case of RT_ExternalMciMovie**	
00005D57	0008	RecordHeader **- rh**	
00005D57	4 bits	unsigned integer **- recVer**	0xF
00005D57	12 bits	unsigned integer **- recInstance**	0x000
00005D59	0002	RecordType **- recType**	0x1007
00005D5B	0004	unsigned integer **- recLen**	0x00000036
00005D5F	0036	ExVideoContainer **- exVideo**	
00005D5F	0008	RecordHeader **- rh**	
00005D67	0010	ExMediaAtom **- exMediaAtom**	
00005D67	0008	RecordHeader **- rh**	
00005D6F	0004	unsigned integer **- exObjId**	0x0000001B
00005D73	0002	unsigned integer **- fLoop**	0x0000
00005D73	0002	unsigned integer **- fRewind**	0x0000
00005D73	0002	unsigned integer **- fNarration**	0x0000
00005D73	0002	unsigned integer **- reserved**	0x0000
00005D77	001E	UncOrLocalPathAtom **- videoFilePathAtom**	
00005D77	0008	RecordHeader **- rh**	
00005D7F	0016	UncOrLocalPath **- path**	C:\Bear.wmv

Figure 40: ExMCIMovieContainer record H child-record hierarchy

exVideo.exMediaAtom.exObjId: 0x0000001B specifies the identifier of the external object. It matches the value of the **exObjRefAtom.exObjIdRef** field in the table titled "OfficeArtSpContainer record A child-record hierarchy" in this section.

exVideo.exMediaAtom.fLoop: 0x0000 specifies that the video data is not repeated continuously.

exVideo.exMediaAtom.fRewind: 0x0000 specifies that the video data is not rewound after playing.

exVideo.exMediaAtom.fNarration: 0x0000 specifies that the video data is not recorded narration for the slide show.

exVideo.videoFilePathAtom.path: "C:\Bear.wmv" specifies the local path string to the video file.

3.9 Text Example

The following sections provide examples of paragraph-level text formatting, character-level text formatting, interactive text, and text metacharacters.

3.9.1 Paragraph Formatting Example

Paragraph-level formatting is stored separately from the text characters.

The most basic paragraph-level formatting for a body of text is specified in the StyleTextPropAtom record. The following sample shows a section of the **SlideListWithTextContainer** record (section 2.4.14.3) pertaining to presentation slide 1 and presentation slide 2. (The complete hierarchy of the **SlideListWithTextContainer** record is shown in the table titled "SlideListWithTextContainer child-record hierarchy" in section 3.4.)

Offset	Size	Structure	Value
000066A0	001C	SlidePersistAtom **- case of RT_SlidePersistAtom A**	
000066BC	000C	TextHeaderAtom **- case of RT_TextHeaderAtom A**	
000066C8	0010	TextBytesAtom **- case of RT_TextBytesAtom A**	
000066D8	000C	TextHeaderAtom **- case of RT_TextHeaderAtom B**	
000066E4	0014	TextBytesAtom **- case of RT_TextBytesAtom B**	
000066F8	001C	SlidePersistAtom **- case of RT_SlidePersistAtom B**	
00006714	000C	TextHeaderAtom **- case of RT_TextHeaderAtom C**	
00006714	0008	RecordHeader **- rh**	
00006714	4 bits	unsigned integer **- recVer**	0x0
00006714	12 bits	unsigned integer **- recInstance**	0x000
00006716	0002	RecordType **- recType**	0x0F9F
00006718	0004	unsigned integer **- recLen**	0x00000004
00006720	0013	TextBytesAtom **- case of RT_TextBytesAtom C**	
00006720	0008	RecordHeader **- rh**	
00006728	000B	array of bytes **- textBytes**	the weather
00006733	000C	TextHeaderAtom **- case of RT_TextHeaderAtom D**	
00006733	0008	RecordHeader **- rh**	
00006733	4 bits	unsigned integer **- recVer**	0x0
00006733	12 bits	unsigned integer **- recInstance**	0x001
00006735	0002	RecordType **- recType**	0x0F9F
00006737	0004	unsigned integer **- recLen**	0x00000004
0000673B	0004	TextTypeEnum **- textType**	0x00000001
0000673F	0031	TextBytesAtom **- case of RT_TextBytesAtom D**	
0000673F	0008	RecordHeader **- rh**	
0000673F	4 bits	unsigned integer **- recVer**	0x0
0000673F	12 bits	unsigned integer **- recInstance**	0x000

Offset	Size	Structure	Value
00006741	0002	RecordType **- recType**	0x0FA8
00006743	0004	unsigned integer **- recLen**	0x00000029
00006747	0029	array of bytes **- textBytes**	a sunny day\rthe blue sky\rsome green grass
00006770	0022	**A**: StyleTextPropAtom **- styleTextPropAtom**	

Figure 41: Basic paragraph-level formatting for placeholder text

The preceding sample shows how the text and basic paragraph-level formatting for the content placeholder on presentation slide 2 of the sample presentation are specified.

case of RT_SlidePersistAtom B: The second **SlidePersistAtom** record (section 2.4.14.5) in the **SlideListWithTextContainer** record specifies presentation slide 2. The records that follow specify the text for placeholders on this slide.

case of RT_TextHeaderAtom C: Specifies the title placeholder shape. The **rh.recInstance** value of 0x000 specifies that this text is the first text body on presentation slide 2.

case of RT_TextHeaderAtom D: Specifies a new body of text. The **rh.recInstance** value of 0x001 specifies that this text is the second text body for presentation slide 2. The records that follow specify additional properties for this text body.

case of RT_TextHeaderAtom D.textType: 0x00000001 specifies that the text type is Tx_TYPE_BODY, the body placeholder shape text.

case of RT_TextBytesAtom D: Specifies the characters of the text. The quantity of characters specified is 41. Therefore, the length of the text is 42 because of the terminating line break character.

styleTextPropAtom: Specifies the text formatting.

The child-record hierarchy of the StyleTextPropAtom record A from the previous table is shown expanded in the following table.

Offset	Size	Structure
00006770	0022	StyleTextPropAtom **- case of RT_StyleTextPropAtom**
00006770	0008	RecordHeader **- rh**
00006778	0012	**A**: TextPFRun **- textPFRun**
0000678A	0008	TextCFRun **- textCFRun**

Figure 42: StyleTextPropAtom child-record hierarchy

textPFRun: Specifies the paragraph-level formatting for a single range of text. There is only one TextPFRun in the StyleTextPropAtom record because all of the individual bulleted paragraphs share the same paragraph-level properties.

The child-record hierarchy of the TextPFRun record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00006778	0012	TextPFRun **- textPFRun**	

Offset	Size	Structure	Value
00006778	0004	unsigned integer **- count**	0x0000002A
0000677C	0002	unsigned integer **- indentLevel**	0x0000
0000677E	000C	TextPFException **- pf**	
0000677E	0004	**A**: PFMasks **- masks**	
00006782	0002	**B**: BulletFlags **- bulletFlags**	
00006784	0002	signed integer **- bulletSize**	0x0032
00006786	0004	ColorIndexStruct **- bulletColor**	

Figure 43: TextPFRun child-record hierarchy

count: 0x0000002A specifies that **textPFRun** applies to 42 characters.

indentLevel: 0x0000 specifies the indent level of the paragraphs.

pf: Specifies the basic paragraph-level formatting options for this run.

pf.masks: Specifies which fields of **pf** exist or are valid.

pf.bulletFlags: Specifies whether the run has certain paragraph-level properties. This field exists because **masks.bulletHasSize** is set. The **masks** field is expanded in the following table.

pf.bulletSize: 0x0032 specifies the bullet size. This field exists because **masks.bulletSize** is set.

pf.bulletColor: Specifies the bullet color. This field exists because **masks.bulletColor** is set.

The child-record hierarchy of the PFMasks record A from the table titled "TextPFRun child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
0000677E	0004	PFMasks **- masks**	
0000677E	1 bit	unsigned integer **- hasBullet**	0x0
0000677E	1 bit	unsigned integer **- bulletHasFont**	0x0
0000677E	1 bit	unsigned integer **- bulletHasColor**	0x0
0000677E	1 bit	unsigned integer **- bulletHasSize**	0x1
0000677E	1 bit	unsigned integer **- bulletFont**	0x0
0000677E	1 bit	unsigned integer **- bulletColor**	0x1
0000677E	1 bit	unsigned integer **- bulletSize**	0x1
0000677E	1 bit	unsigned integer **- bulletChar**	0x0
0000677E	1 bit	unsigned integer **- leftMargin**	0x0
0000677E	1 bit	unsigned integer **- unused**	0x0
0000677E	1 bit	unsigned integer **- indent**	0x0
0000677E	1 bit	unsigned integer **- align**	0x0

Offset	Size	Structure	Value
0000677E	1 bit	unsigned integer **- lineSpacing**	0x0
0000677E	1 bit	unsigned integer **- spaceBefore**	0x0
0000677E	1 bit	unsigned integer **- spaceAfter**	0x0
0000677E	1 bit	unsigned integer **- defaultTabSize**	0x0
0000677E	1 bit	unsigned integer **- fontAlign**	0x0
0000677E	1 bit	unsigned integer **- charWrap**	0x0
0000677E	1 bit	unsigned integer **- wordWrap**	0x0
0000677E	1 bit	unsigned integer **- overflow**	0x0
0000677E	1 bit	unsigned integer **- tabStops**	0x0
0000677E	1 bit	unsigned integer **- textDirection**	0x0
0000677E	1 bit	unsigned integer **- reserved1**	0x0
0000677E	1 bit	unsigned integer **- bulletBlip**	0x0
0000677E	1 bit	unsigned integer **- bulletScheme**	0x0
0000677E	1 bit	unsigned integer **- bulletHasScheme**	0x0
0000677E	6 bits	unsigned integer **- reserved2**	0x00

Figure 44: PFMasks contents

bulletHasSize: 0x00000001 specifies that the **bulletFlags** field of the TextPFException shown in the table titled "TextPFRun child-record hierarchy" in this section exists.

bulletColor: 0x00000001 specifies that the **bulletColor** field of the TextPFException shown in the table titled "TextPFRun child-record hierarchy" in this section exists.

bulletSize: 0x00000001 specifies that the **bulletSize** field of the TextPFException shown in the table titled "TextPFRun child-record hierarchy" in this section exists.

The child-record hierarchy of the BulletFlags record B from the table titled "TextPFRun child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00006782	0002	BulletFlags **- bulletFlags**	
00006782	1 bit	unsigned integer **- fHasBullet**	0x1
00006782	1 bit	unsigned integer **- fBulletHasFont**	0x0
00006782	1 bit	unsigned integer **- fBulletHasColor**	0x0
00006782	1 bit	unsigned integer **- fBulletHasSize**	0x1
00006782	12 bits	unsigned integer **- reserved**	0x000

Figure 45: BulletFlags contents

fHasBullet: 0x0001 specifies that a bullet exists.

fBulletHasSize: 0x0001 specifies that the **bulletSize** field of the TextPFException shown in the table titled "TextPFRun child-record hierarchy" in this section is valid.

The following example shows how additional paragraph-level formatting is specified. The following OutlineTextProps2.9.61Entry record is located in the OutlineTextProps2.9.60Container record contained in the PP2.4.23.5DocBinaryTagExtension record shown in the table titled "PP9DocBinaryTagExtension child-record hierarchy" in section 3.6.1.

Offset	Size	Structure	Value
0000664A	0026	OutlineTextProps9Entry **- outlineTextProps9Entry**	
0000664A	0010	OutlineTextPropsHeaderExAtom **- outlineTextHeaderAtom**	
0000664A	0008	RecordHeader **- rh**	
0000664A	4 bits	unsigned integer **- recVer**	0x0
0000664A	12 bits	unsigned integer **- recInstance**	0x001
0000664C	0002	RecordType **- recType**	0x0FAF
0000664E	0004	unsigned integer **- recLen**	0x00000008
00006652	0004	SlideIdRef **- slideIdRef**	0x00000101
00006656	0004	TextTypeEnum **- txType**	0x00000001
0000665A	0016	StyleTextProp9Atom **- styleTextProp9Atom**	
0000665A	0008	RecordHeader **- rh**	
00006662	000E	StyleTextProp9 **- styleTextProps9**	
00006662	0006	TextPFException9 **- pf9**	
00006662	0004	PFMasks **- masks**	
00006666	0002	signed integer **- bulletBlipRef**	0x0000
00006668	0004	TextCFException9 **- cf9**	
0000666C	0004	TextSIException **- si**	

Figure 46: An OutlineTextProps9Entry that contains the additional paragraph-level properties for the content placeholder on presentation slide 2

The OutlineTextPropsHeaderExAtom specifies the corresponding text to which this formatting applies. The **rh.recInstance** field of the OutlineTextPropsHeaderExAtom along with the **slideIdRef** and **txType** fields are sufficient to find the corresponding text specified by a TextHeaderAtom record contained in the **SlideListWithTextContainer** record.

The OutlineTextProps2.9.61Entry contains one StyleTextProp2.9.67Atom because there is only one run in the text. This is the case because all bits of the **fontStyle.pp9rt** fields of all TextCFException records contained in the TextCFRun records are equal. Therefore, the TextPFException2.9.26, TextCFException2.9.17, and TextSIException apply to the entire text. The TextCFException2.9.17 specifies that there is a custom graphical bullet for this placeholder.

outlineTextHeaderAtom.rh.recInstance: 0x001 specifies that these additional properties apply to the second text body for the presentation slide specified by the **slideIdRef** field.

outlineTextHeaderAtom.slideIdRef: 0x00000101 specifies the presentation slide that contains the text that has additional paragraph-level properties.

outlineTextHeaderAtom.txType: 0x00000001 specifies the type of text. This matches the **txType** field of the TextHeaderAtom shown in the table titled "Basic paragraph-level formatting for placeholder text" in this section.

styleTextProp9Atom: Specifies the additional paragraph-level properties.

styleTextProp9Atom.styleTextProps9.pf9.bulletBlipRef: 0x0000 specifies the picture to use as the bullet symbol.

3.9.2 Character Formatting Example

This example shows how character-level formatting is specified for the green circular shape on presentation slide 6 as shown in figure titled "Presentation slide 6" in section 3.1.

The text of the shape is specified in an OfficeArtClientTextbox record. The child-record hierarchy of the **OfficeArtSpContainer** ([MS-ODRAW] section 2.2.14) record D from the table titled "DrawingContainer child-record hierarchy" in section 3.8.1 is shown expanded in the following table.

Offset	Size	Structure	Value
00007E30	00FB	OfficeArtSpContainer **- case of msofbtSpContainer**	
00007E30	0008	OfficeArtRecordHeader **- rh**	
00007E38	0010	OfficeArtFSP **- shapeProp**	
00007E48	0044	OfficeArtFOPT **- shapePrimaryOptions**	
00007E8C	000E	OfficeArtTertiaryFOPT **- shapeTertiaryOptions**	
00007E9A	0010	OfficeArtClientAnchor **- clientAnchor**	
00007EAA	0081	OfficeArtClientTextbox **- clientTextbox**	
00007EAA	0008	OfficeArtRecordHeader **- rh**	
00007EB2	000C	TextHeaderAtom **- case of RT_TextHeaderAtom**	
00007EBE	001D	TextBytesAtom **- case of RT_TextBytesAtom**	
00007EBE	0008	RecordHeader **- rh**	
00007EBE	4 bits	unsigned integer **- recVer**	0x0
00007EBE	12 bits	unsigned integer **- recInstance**	0x000
00007EC0	0002	RecordType **- recType**	0x0FA8
00007EC2	0004	unsigned integer **- recLen**	0x00000015
00007EC6	0015	array of bytes **- textBytes**	an \runderlined\rcircle
00007EDB	0050	StyleTextPropAtom **- case of RT_StyleTextPropAtom**	
00007EDB	0008	RecordHeader **- rh**	
00007EE3	000C	TextPFRun **- textPFRun**	
00007EEF	000E	**A**: TextCFRun **- TextCFRun A**	
00007EFD	0010	**B**: TextCFRun **- TextCFRun B**	

Offset	Size	Structure	Value
00007F0D	000E	**C**: TextCFRun **- TextCFRun C**	
00007F1B	0010	**D**: TextCFRun **- TextCFRun D**	

Figure 47: OfficeArtSpContainer child-record hierarchy

clientTextbox.case of RT_TextBytesAtom.rh.recLen: 0x00000015 specifies that the length of the corresponding text body is 22 because of the terminating line break character.

clientTextbox.case of RT_TextBytesAtom.textBytes: "an \runderlined\rcircle" specifies the content of the text.

The basic character-level formatting is specified by four TextCFRun structures contained in the StyleTextPropAtom record that follows the TextBytesAtom record.

The child-record hierarchy of the TextCFRun record A from the previous table is shown expanded in the following table. The offset of the first TextCFRun depends on the paragraph-level formatting. When all TextPFRun records are read, the character-level formatting begins.

Offset	Size	Structure	Value
00007EEF	000E	**A**: TextCFRun **- textCFRun**	
00007EEF	0004	unsigned integer **- count**	0x00000002
00007EF3	000A	TextCFException **- cf**	
00007EF3	0004	CFMasks **- masks**	
00007EF3	1 bit	unsigned integer **- bold**	0x0
00007EF3	1 bit	unsigned integer **- italic**	0x0
00007EF3	1 bit	unsigned integer **- underline**	0x0
00007EF3	1 bit	unsigned integer **- unused1**	0x0
00007EF3	1 bit	unsigned integer **- shadow**	0x0
00007EF3	1 bit	unsigned integer **- fehint**	0x0
00007EF3	1 bit	unsigned integer **- unused2**	0x0
00007EF3	1 bit	unsigned integer **- kumi**	0x0
00007EF3	1 bit	unsigned integer **- unused3**	0x0
00007EF3	1 bit	unsigned integer **- emboss**	0x0
00007EF3	4 bits	unsigned integer **- fHasStyle**	0x0
00007EF3	2 bits	unsigned integer **- unused4**	0x0
00007EF3	1 bit	unsigned integer **- typeface**	0x0
00007EF3	1 bit	unsigned integer **- size**	0x0
00007EF3	1 bit	unsigned integer **- color**	0x1
00007EF3	1 bit	unsigned integer **- position**	0x0

Offset	Size	Structure	Value
00007EF3	1 bit	unsigned integer **- pp10ext**	0x0
00007EF3	1 bit	unsigned integer **- oldEATypeface**	0x0
00007EF3	1 bit	unsigned integer **- ansiTypeface**	0x1
00007EF3	1 bit	unsigned integer **- symbolTypeface**	0x0
00007EF3	1 bit	unsigned integer **- newEATypeface**	0x0
00007EF3	1 bit	unsigned integer **- csTypeface**	0x0
00007EF3	1 bit	unsigned integer **- pp11ext**	0x0
00007EF3	5 bits	unsigned integer **- reserved**	0x00
00007EF7	0002	unsigned integer **- ansiFontRef**	0x0000
00007EF9	0004	ColorIndexStruct **- color**	

Figure 48: TextCFRun record A child-record hierarchy

This TextCFRun specifies only the font and font color for the first two characters, "an". The **fontStyle** field does not exist, because none of the bits of the CFMasks record that determine the existence of **fontStyle** are set.

count: 0x00000002 specifies that these character-level formatting options apply to two characters. The corresponding characters are "an".

cf.masks.color: 0x0001 specifies that **cf.color** exists.

cf.masks.ansiTypeface: 0x0001 specifies that **cf.ansiFontRef** exists.

cf.ansiFontRef: 0x0000 specifies the font. This field exists because **cf.masks.ansiTypeface** is set.

cf.color: Specifies the color of the font. This field exists because **cf.masks.color** is set.

So far, the sum of the **count** field of the TextCFRun records is 2.

The child-record hierarchy of the TextCFRun record B from the table titled "OfficeArtSpContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00007EFD	0010	**B**: TextCFRun **- textCFRun**	
00007EFD	0004	unsigned integer **- count**	0x00000002
00007F01	000C	TextCFException **- cf**	
00007F01	0004	CFMasks **- masks**	
00007F01	1 bit	unsigned integer **- bold**	0x0
00007F01	1 bit	unsigned integer **- italic**	0x0
00007F01	1 bit	unsigned integer **- underline**	0x1
00007F01	1 bit	unsigned integer **- unused1**	0x0
00007F01	1 bit	unsigned integer **- shadow**	0x0

Offset	Size	Structure	Value
00007F01	1 bit	unsigned integer **- fehint**	0x0
00007F01	1 bit	unsigned integer **- unused2**	0x0
00007F01	1 bit	unsigned integer **- kumi**	0x0
00007F01	1 bit	unsigned integer **- unused3**	0x0
00007F01	1 bit	unsigned integer **- emboss**	0x0
00007F01	4 bits	unsigned integer **- fHasStyle**	0x0
00007F01	2 bits	unsigned integer **- unused4**	0x0
00007F01	1 bit	unsigned integer **- typeface**	0x0
00007F01	1 bit	unsigned integer **- size**	0x0
00007F01	1 bit	unsigned integer **- color**	0x1
00007F01	1 bit	unsigned integer **- position**	0x0
00007F01	1 bit	unsigned integer **- pp10ext**	0x0
00007F01	1 bit	unsigned integer **- oldEATypeface**	0x0
00007F01	1 bit	unsigned integer **- ansiTypeface**	0x1
00007F01	1 bit	unsigned integer **- symbolTypeface**	0x0
00007F01	1 bit	unsigned integer **- newEATypeface**	0x0
00007F01	1 bit	unsigned integer **- csTypeface**	0x0
00007F01	1 bit	unsigned integer **- pp11ext**	0x0
00007F01	5 bits	unsigned integer **- reserved**	0x00
00007F05	0002	CFStyle **- fontStyle**	
00007F05	1 bit	unsigned integer **- bold**	0x0
00007F05	1 bit	unsigned integer **- italic**	0x0
00007F05	1 bit	unsigned integer **- underline**	0x1
00007F05	1 bit	unsigned integer **- unused1**	0x0
00007F05	1 bit	unsigned integer **- shadow**	0x0
00007F05	1 bit	unsigned integer **- fehint**	0x0
00007F05	1 bit	unsigned integer **- unused2**	0x0
00007F05	1 bit	unsigned integer **- kumi**	0x0
00007F05	1 bit	unsigned integer **- unused3**	0x0
00007F05	1 bit	unsigned integer **- emboss**	0x0
00007F05	4 bits	unsigned integer **- pp9rt**	0x0

Offset	Size	Structure	Value
00007F05	2 bits	unsigned integer **- unused4**	0x0
00007F07	0002	unsigned integer **- ansiFontRef**	0x0000
00007F09	0004	ColorIndexStruct **- color**	

Figure 49: TextCFRun B child-record hierarchy

count: 0x00000002 specifies that these character-level formatting options apply to the next 2 characters, " \r".

cf.masks.underline: 0x0001 specifies that **cf.fontStyle** exists and that **cf.fontStyle.underline** is valid.

cf.fontStyle: Specifies styling options. This field exists because **cf.masks.underline** is set.

cf.fontStyle.underline: 0x0001 specifies that the text is underlined.

So far, the sum of the **count** field of the TextCFRun records is 4.

The child-record hierarchy of the TextCFRun record C from the table titled "OfficeArtSpContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00007F0D	000E	**C**: TextCFRun **- textCFRun**	
00007F0D	0004	unsigned integer **- count**	0x0000000B
00007F11	000A	TextCFException **- cf**	
00007F11	0004	CFMasks **- masks**	
00007F11	1 bit	unsigned integer **- bold**	0x0
00007F11	1 bit	unsigned integer **- italic**	0x0
00007F11	1 bit	unsigned integer **- underline**	0x0
00007F11	1 bit	unsigned integer **- unused1**	0x0
00007F11	1 bit	unsigned integer **- shadow**	0x0
00007F11	1 bit	unsigned integer **- fehint**	0x0
00007F11	1 bit	unsigned integer **- unused2**	0x0
00007F11	1 bit	unsigned integer **- kumi**	0x0
00007F11	1 bit	unsigned integer **- unused3**	0x0
00007F11	1 bit	unsigned integer **- emboss**	0x0
00007F11	4 bits	unsigned integer **- fHasStyle**	0x0
00007F11	2 bits	unsigned integer **- unused4**	0x0
00007F11	1 bit	unsigned integer **- typeface**	0x0
00007F11	1 bit	unsigned integer **- size**	0x0

Offset	Size	Structure	Value
00007F11	1 bit	unsigned integer **- color**	0x1
00007F11	1 bit	unsigned integer **- position**	0x0
00007F11	1 bit	unsigned integer **- pp10ext**	0x0
00007F11	1 bit	unsigned integer **- oldEATypeface**	0x0
00007F11	1 bit	unsigned integer **- ansiTypeface**	0x1
00007F11	1 bit	unsigned integer **- symbolTypeface**	0x0
00007F11	1 bit	unsigned integer **- newEATypeface**	0x0
00007F11	1 bit	unsigned integer **- csTypeface**	0x0
00007F11	1 bit	unsigned integer **- pp11ext**	0x0
00007F11	5 bits	unsigned integer **- reserved**	0x00
00007F15	0002	unsigned integer **- ansiFontRef**	0x0000
00007F17	0004	ColorIndexStruct **- color**	

Figure 50: TextCFRun C child-record hierarchy

The **fontStyle** field does not exist because none of the first 16 bits of the CFMasks record is set.

count: 0x0000000B specifies that these character-level properties apply to the next 11 characters, "underlined\r".

cf.masks.underline: 0x0000 specifies that the **underline** field of **fontStyle**, which does not exist in this TextCFRun, would not be valid.

So far, the sum of the **count** field of the TextCFRun records is 15.

The child-record hierarchy of the TextCFRun record D from the table titled "OfficeArtSpContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00007F1B	0010	**D**: TextCFRun **- textCFRun**	
00007F1B	0004	unsigned integer **- count**	0x00000007
00007F1F	000C	TextCFException **- cf**	
00007F1F	0004	CFMasks **- masks**	
00007F1F	1 bit	unsigned integer **- bold**	0x0
00007F1F	1 bit	unsigned integer **- italic**	0x0
00007F1F	1 bit	unsigned integer **- underline**	0x1
00007F1F	1 bit	unsigned integer **- unused1**	0x0
00007F1F	1 bit	unsigned integer **- shadow**	0x0
00007F1F	1 bit	unsigned integer **- fehint**	0x0

Offset	Size	Structure	Value
00007F1F	1 bit	unsigned integer **- unused2**	0x0
00007F1F	1 bit	unsigned integer **- kumi**	0x0
00007F1F	1 bit	unsigned integer **- unused3**	0x0
00007F1F	1 bit	unsigned integer **- emboss**	0x0
00007F1F	4 bits	unsigned integer **- fHasStyle**	0x0
00007F1F	2 bits	unsigned integer **- unused4**	0x0
00007F1F	1 bit	unsigned integer **- typeface**	0x0
00007F1F	1 bit	unsigned integer **- size**	0x0
00007F1F	1 bit	unsigned integer **- color**	0x1
00007F1F	1 bit	unsigned integer **- position**	0x0
00007F1F	1 bit	unsigned integer **- pp10ext**	0x0
00007F1F	1 bit	unsigned integer **- oldEATypeface**	0x0
00007F1F	1 bit	unsigned integer **- ansiTypeface**	0x1
00007F1F	1 bit	unsigned integer **- symbolTypeface**	0x0
00007F1F	1 bit	unsigned integer **- newEATypeface**	0x0
00007F1F	1 bit	unsigned integer **- csTypeface**	0x0
00007F1F	1 bit	unsigned integer **- pp11ext**	0x0
00007F1F	5 bits	unsigned integer **- reserved**	0x00
00007F23	0002	CFStyle **- fontStyle**	
00007F23	1 bit	unsigned integer **- bold**	0x0
00007F23	1 bit	unsigned integer **- italic**	0x0
00007F23	1 bit	unsigned integer **- underline**	0x1
00007F23	1 bit	unsigned integer **- unused1**	0x0
00007F23	1 bit	unsigned integer **- shadow**	0x0
00007F23	1 bit	unsigned integer **- fehint**	0x0
00007F23	1 bit	unsigned integer **- unused2**	0x0
00007F23	1 bit	unsigned integer **- kumi**	0x0
00007F23	1 bit	unsigned integer **- unused3**	0x0
00007F23	1 bit	unsigned integer **- emboss**	0x0
00007F23	4 bits	unsigned integer **- pp9rt**	0x0
00007F23	2 bits	unsigned integer **- unused4**	0x0

Offset	Size	Structure	Value
00007F25	0002	unsigned integer **- ansiFontRef**	0x0000
00007F27	0004	ColorIndexStruct **- color**	

Figure 51: TextCFRun D child-record hierarchy

count: 0x00000007 specifies that these character-level properties apply to the next 7 characters, "circle" and the terminating '\n'. Note that the terminating '\n' is not included in the preceding TextBytesAtom shown in the table titled "OfficeArtSpContainer child-record hierarchy" in this section.

cf.masks.underline: 0x0001 specifies that **cf.fontStyle** exists and that **cf.fontStyle.underline** is valid.

cf.fontStyle: Specifies character-level styling.

cf.fontStyle.underline: 0x0001 specifies that the text is underlined.

So far, the sum of the **count** field of the TextCFRun text structures is 22. Because the sum of the **count** field is the length of the text, this TextCFRun record is the final TextCFRun record.

3.9.3 TextInteractiveInfo Example

This example shows how the hyperlink is specified for the yellow square on presentation slide 6 as shown in figure titled "Presentation slide 6" in section 3.1.

The clickable text hyperlink is specified by an MouseClickInteractiveInfoContainer record and a MouseClickTextInteractiveInfoAtom record following the TextHeaderAtom record, the characters of the text, and the formatting.

The child-record hierarchy of the **OfficeArtSpContainer** ([MS-ODRAW] section 2.2.14) record B from the table titled "DrawingContainer child-record hierarchy" in section 3.8.1 is shown expanded in the following table.

Offset	Size	Structure	Value
00007C30	0109	OfficeArtSpContainer **- case of msofbtSpContainer**	
00007C30	0008	OfficeArtRecordHeader **- rh**	
00007C38	0010	OfficeArtFSP **- shapeProp**	
00007C48	0044	OfficeArtFOPT **- shapePrimaryOptions**	
00007C8C	000E	OfficeArtTertiaryFOPT **- shapeTertiaryOptions**	
00007C9A	0010	OfficeArtClientAnchor **- clientAnchor**	
00007CAA	008F	OfficeArtClientTextbox **- clientTextbox**	
00007CAA	0008	OfficeArtRecordHeader **- rh**	
00007CB2	000C	TextHeaderAtom **- case of RT_TextHeaderAtom**	
00007CBE	001D	TextBytesAtom **- case of RT_TextBytesAtom**	
00007CBE	0008	RecordHeader **- rh**	
00007CC6	0015	array of bytes **- textBytes**	a \rhyperlinked\rsquare

Offset	Size	Structure	Value
00007CDB	002E	StyleTextPropAtom **- case of RT_StyleTextPropAtom**	
00007D09	0020	**A**: MouseClickInteractiveInfoContainer **- mouseClickInteractiveInfoContainer**	
00007D29	0010	**B**: MouseClickTextInteractiveInfoAtom **- mouseClickTextInteractiveInfoAtom**	

Figure 52: OfficeArtSpContainer child-record hierarchy

The child-record hierarchy of the MouseClickInteractiveInfoContainer record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
00007D09	0020	MouseClickInteractiveInfoContainer **- mouseClickInteractiveInfoContainer**	
00007D09	0008	RecordHeader **- rh**	
00007D09	4 bits	unsigned integer **- recVer**	0xF
00007D09	12 bits	unsigned integer **- recInstance**	0x000
00007D0B	0002	RecordType **- recType**	0x0FF2
00007D0D	0004	unsigned integer **- recLen**	0x00000018
00007D11	0018	InteractiveInfoAtom **- interactiveInfoAtom**	
00007D11	0008	RecordHeader **- rh**	
00007D11	4 bits	unsigned integer **- recVer**	0x0
00007D11	12 bits	unsigned integer **- recInstance**	0x000
00007D13	0002	RecordType **- recType**	0x0FF3
00007D15	0004	unsigned integer **- recLen**	0x00000010
00007D19	0004	unsigned integer **- soundIdRef**	0x00000000
00007D1D	0004	unsigned integer **- exHyperlinkIdRef**	0x0000001C
00007D21	0001	InteractiveInfoActionEnum **- action**	0x04
00007D22	0001	OLEVerbEnum **- oleVerb**	0x00
00007D23	0001	InteractiveInfoJumpEnum **- jump**	0x00
00007D24	1 bit	bit **- fAnimated**	0x0
00007D24	1 bit	bit **- fStopSound**	0x0
00007D24	1 bit	bit **- fCustomShowReturn**	0x0
00007D24	1 bit	bit **- fVisited**	0x0
00007D24	4 bits	unsigned integer **- reserved**	0x0
00007D25	0001	LinkToEnum **- hyperlinkType**	0x08

Figure 53: MouseClickInteractiveInfoContainer child-record hierarchy

interactiveInfoAtom.exHyperlinkIdRef: 0x0000001C specifies the identifier of the hyperlink that is the target of this interactive portion of text.

The child-record hierarchy of the MouseClickTextInteractiveInfoAtom record B from the table titled "OfficeArtSpContainer child-record hierarchy" in this section is shown expanded in the following table.

Offset	Size	Structure	Value
00007D29	0010	MouseClickTextInteractiveInfoAtom **- mouseClickTextInteractiveInfoAtom**	
00007D29	0008	RecordHeader **- rh**	
00007D29	4 bits	unsigned integer **- recVer**	0x0
00007D29	12 bits	unsigned integer **- recInstance**	0x000
00007D2B	0002	RecordType **- recType**	0x0FDF
00007D2D	0004	unsigned integer **- recLen**	0x00000008
00007D31	0008	TextRange **- range**	
00007D31	0004	signed integer **- begin**	0x0000000F
00007D35	0004	signed integer **- end**	0x00000015

Figure 54: MouseClickTextInteractiveInfoAtom child-record hierarchy

range.begin: 0x0000000F specifies the zero-based index of the first character, the 's', to which this hyperlink applies.

range.end: 0x00000015 specifies the zero-based index of the last character, the 'e', to which this hyperlink applies.

An ExHyperlinkContainer record exists for each hyperlink referenced in the document. The following ExHyperlinkContainer record is the one referenced in the preceding sample of the yellow square.

The child-record hierarchy of the ExHyperlinkContainer record H from the table titled "ExObjListContainer child-record hierarchy" in section 3.8.4 is shown expanded in the following table.

Offset	Size	Structure	Value
00005D95	005E	ExHyperlinkContainer **- case of RT_ExternalHyperlink**	
00005D95	0008	RecordHeader **- rh**	
00005D9D	000C	ExHyperlinkAtom **- exHyperlinkAtom**	
00005D9D	0008	RecordHeader **- rh**	
00005DA5	0004	unsigned integer **- exHyperlinkId**	0x0000001C
00005DA9	0014	FriendlyNameAtom **- friendlyNameAtom**	
00005DA9	0008	RecordHeader **- rh**	
00005DB1	000C	array of bytes **- friendlyName**	square
00005DBD	0036	TargetAtom **- targetAtom**	
00005DBD	0008	RecordHeader **- rh**	

Offset	Size	Structure	Value
00005DC5	002E	array of bytes **- target**	http://www.contoso.com/

Figure 55: ExHyperlinkContainer child-record hierarchy

The **exHyperlinkAtom.exHyperlinkId** field matches the **exHyperlinkIdRef** field of the InteractiveInfoAtom record shown in the table titled "MouseClickInteractiveInfoContainer child-record hierarchy" in this section.

targetAtom: Specifies the target of the hyperlink. In this case, it is the external Web site located at http://www.contoso.com/.

3.9.4 Metacharacter Example

The following example shows how a footer metacharacter is specified on presentation slide 6.

The following diagram shows the context of a footer placeholder on the main master slide.

The child-record hierarchy of the **MainMasterContainer** (section 2.5.3) record B from the table titled "Top-level record sequence in the PowerPoint Document Stream from sample.ppt" in section 3.3 is shown expanded in the following table.

Offset	Size	Structure
00000CC5	09AC	MainMasterContainer **- case of RT_MainMaster**
00000CC5	0008	RecordHeader **- rh**
00000CCD	0020	SlideAtom **- slideAtom**
00000CED	0028	SchemeListElementColorSchemeAtom **- schemeListElementColorSchemeAtom**
00000D15	0028	SchemeListElementColorSchemeAtom **- schemeListElementColorSchemeAtom**
00000D3D	0028	SchemeListElementColorSchemeAtom **- schemeListElementColorSchemeAtom**
00000D65	0028	SchemeListElementColorSchemeAtom **- schemeListElementColorSchemeAtom**
00000D8D	0028	SchemeListElementColorSchemeAtom **- schemeListElementColorSchemeAtom**
00000DB5	0028	SchemeListElementColorSchemeAtom **- schemeListElementColorSchemeAtom**
00000DDD	0028	SchemeListElementColorSchemeAtom **- schemeListElementColorSchemeAtom**
00000E05	0028	SchemeListElementColorSchemeAtom **- schemeListElementColorSchemeAtom**
00000E2D	0028	SchemeListElementColorSchemeAtom **- schemeListElementColorSchemeAtom**
00000E55	0028	SchemeListElementColorSchemeAtom **- schemeListElementColorSchemeAtom**
00000E7D	0028	SchemeListElementColorSchemeAtom **- schemeListElementColorSchemeAtom**
00000EA5	0028	SchemeListElementColorSchemeAtom **- schemeListElementColorSchemeAtom**
00000ECD	0046	TextMasterStyleAtom **- textMasterStyleAtom**
00000F13	0084	TextMasterStyleAtom **- textMasterStyleAtom**
00000F97	0076	TextMasterStyleAtom **- textMasterStyleAtom**
0000100D	005A	TextMasterStyleAtom **- textMasterStyleAtom**

Offset	Size	Structure
00001067	0014	TextMasterStyleAtom **- textMasterStyleAtom**
0000107B	0046	TextMasterStyleAtom **- textMasterStyleAtom**
000010C1	0046	TextMasterStyleAtom **- textMasterStyleAtom**
00001107	04DE	DrawingContainer **- drawing**
00001107	0008	RecordHeader **- rh**
0000110F	04D6	OfficeArtDgContainer **- OfficeArtDg**
0000110F	0008	OfficeArtRecordHeader **- rh**
00001117	0010	OfficeArtFDG **- drawingData**
00001127	046E	**A**: OfficeArtSpgrContainer **- groupShape**
00001595	0050	OfficeArtSpContainer **- shape**
000015E5	0028	SlideSchemeColorSchemeAtom **- slideSchemeColorSchemeAtom**
0000160D	0040	SlideProgTagsContainer **- slideProgTagsContainer**
0000164D	0024	TemplateNameAtom **- templateNameAtom**

Figure 56: MainMasterContainer child-record hierarchy

The child-record hierarchy of the **OfficeArtSpgrContainer** record A from the previous table is shown expanded in the following table.

Offset	Size	Structure
00001127	046E	OfficeArtSpgrContainer **- groupShape**
00001127	0008	OfficeArtRecordHeader **- rh**
0000112F	0030	OfficeArtSpContainer **- case of msofbtSpContainer**
0000115F	00DA	OfficeArtSpContainer **- case of msofbtSpContainer**
00001239	011E	OfficeArtSpContainer **- case of msofbtSpContainer**
00001357	00BE	OfficeArtSpContainer **- case of msofbtSpContainer**
00001415	00C0	OfficeArtSpContainer **- case of msofbtSpContainer**
00001415	0008	OfficeArtRecordHeader **- rh**
0000141D	0010	OfficeArtFSP **- shapeProp**
0000142D	0038	OfficeArtFOPT **- shapePrimaryOptions**
00001465	0010	OfficeArtClientAnchor **- clientAnchor**
00001475	0018	OfficeArtClientData **- clientData**
0000148D	0048	**A**: OfficeArtClientTextbox **- clientTextbox**
000014D5	00C0	OfficeArtSpContainer **- case of msofbtSpContainer**

Figure 57: OfficeArtSpgrContainer child-record hierarchy

The child-record hierarchy of the OfficeArtClientTextbox record A from the previous table is shown expanded in the following table.

Offset	Size	Structure	Value
0000148D	0048	OfficeArtClientTextbox **- clientTextbox**	
0000148D	0008	OfficeArtRecordHeader **- rh**	
00001495	000C	TextHeaderAtom **- case of RT_TextHeaderAtom**	
00001495	0008	RecordHeader **- rh**	
0000149D	0004	TextTypeEnum **- textType**	0x00000004
000014A1	000A	TextCharsAtom **- case of RT_TextCharsAtom**	
000014A1	0008	RecordHeader **- rh**	
000014A9	0002	array of bytes **- textChars**	*
000014AB	001E	StyleTextPropAtom **- case of RT_StyleTextPropAtom**	
000014AB	0008	RecordHeader **- rh**	
000014B3	000C	TextPFRun **- textPFRun**	
000014BF	000A	TextCFRun **- textCFRun**	
000014BF	0004	unsigned integer **- count**	0x00000002
000014C3	0006	TextCFException **- cf**	
000014C9	000C	FooterMCAtom **- case of RT_FooterMetaCharAtom**	
000014C9	0008	RecordHeader **- rh**	
000014D1	0004	signed integer **- position**	0x00000000

Figure 58: OfficeArtClientTextbox record A child-record hierarchy

case of RT_TextHeaderAtom.textType: 0x00000004 specifies that this text is of type Tx_TYPE_OTHER.

case of RT_TextCharsAtom.textChars: "*" is only used as a placeholder in this context.

case of RT_FooterMetaCharAtom.position: 0x00000000 specifies that a footer metacharacter exists as the first character in the placeholder.

The main master slide specified by the **MainMasterContainer** record is the main master slide for presentation slide 6. When presentation slide 6 contains a FooterAtom record, the text specified by the FooterAtom is placed into the shape specified by the OfficeArtClientTextBox on the main master slide.

Slide-level footer information is contained in the PerSlideHeadersFootersContainer record contained in the **SlideContainer** (section 2.5.1) for presentation slide 6.

The child-record hierarchy of the PerSlideHeadersFootersContainer record B from the table titled "SlideContainer record U child-record hierarchy" in section 3.5.2 is shown expanded in the following table.

Offset	Size	Structure
00007B1E	003A	PerSlideHeadersFootersContainer **- perSlideHeadersFootersContainer**

Offset	Size	Structure
00007B1E	0008	RecordHeader **- rh**
00007B26	000C	**A**: HeadersFootersAtom **- hfAtom**
00007B32	0026	**B**: FooterAtom **- footerAtom**

Figure 59: PerSlideHeadersFooterContainer child-record hierarchy

The child-record hierarchy of the HeadersFootersAtom record A from this section is shown expanded in the following table.

Offset	Size	Structure	Value
00007B26	000C	**A**: HeadersFootersAtom **- hfAtom**	
00007B26	0008	RecordHeader **- rh**	
00007B2E	0002	signed integer **- formatId**	0x000D
00007B30	1 bit	unsigned integer **- fHasDate**	0x0
00007B30	1 bit	unsigned integer **- fHasTodayDate**	0x0
00007B30	1 bit	unsigned integer **- fHasUserDate**	0x1
00007B30	1 bit	unsigned integer **- fHasSlideNumber**	0x0
00007B30	1 bit	unsigned integer **- fHasHeader**	0x0
00007B30	1 bit	unsigned integer **- fHasFooter**	0x1
00007B30	10 bits	unsigned integer **- reserved**	0x000

Figure 60: HeaderFootersAtom record A child-record hierarchy

fHasFooter: 0x0001 specifies that this slide should show the footer.

The child-record hierarchy of the FooterAtom record B from this section is shown expanded in the following table.

Offset	Size	Structure	Value
00007B32	0026	**B**: FooterAtom **- footerAtom**	
00007B32	0008	RecordHeader **- rh**	
00007B32	4 bits	unsigned integer **- recVer**	0x0
00007B32	12 bits	unsigned integer **- recInstance**	0x002
00007B34	0002	RecordType **- recType**	0x0FBA
00007B36	0004	unsigned integer **- recLen**	0x0000001E
00007B3A	001E	array of bytes **- footer**	text formatting

Figure 61: FooterAtom record B child-record hierarchy

footer: "text formatting" specifies the characters to display for the footer metacharacter when it appears on this slide.

4 Security Considerations

For encrypted documents, the block number 0 is reused for every field of an **OfficeArtBStoreContainerFileBlock** record (described in [MS-ODRAW] section 2.2.22) within the **Pictures Stream**.

Additional security considerations for encrypted documents are described in [MS-OFFCRYPTO] section 4.1.3.2.

See the **CryptSession10Container** record (section 2.3.7) for more information about how to encrypt and decrypt encrypted documents.

5 Appendix A: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs.

- Microsoft PowerPoint 97
- Microsoft PowerPoint 2000
- Microsoft PowerPoint 2002
- Microsoft Office PowerPoint 2003
- Microsoft Office PowerPoint 2007
- Microsoft PowerPoint 2010
- Microsoft PowerPoint 2013
- Microsoft PowerPoint 2016

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. The new behavior also applies to subsequent service packs of the product unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

Unless otherwise specified, any statement of optional behavior in this specification that is prescribed using the terms SHOULD or SHOULD NOT implies product behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that the product does not follow the prescription.

<1> Section 2.1.2: PowerPoint 2000, PowerPoint 2002, Office PowerPoint 2003, Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 preserve unknown records found in the following records: **DocumentContainer** (section 2.4.1), **DrawingContainer** (section 2.5.13), **DrawingGroupContainer** (section 2.4.3), **OfficeArtClientData** (section 2.7.3), **ExControlContainer** (section 2.10.10), **ExOleEmbedContainer** (section 2.10.27), **ExOleLinkContainer** (section 2.10.29), **HandoutContainer** (section 2.5.8), **Kinsoku9Container** (section 2.9.6), **KinsokuContainer** (section 2.9.2), **MainMasterContainer** (section 2.5.3), **MasterListWithTextContainer** (section 2.4.14.1), **NamedShowsContainer** (section 2.6.2), **NormalViewSetInfoContainer** (section 2.4.21.2), **NotesContainer** (section 2.5.6), **NotesListWithTextContainer** (section 2.4.14.6), **NotesViewInfoContainer** (section 2.4.21.12), **SubEffectContainer** (section 2.8.16), **SlideContainer** (section 2.5.1), **SlideListWithTextContainer** (section 2.4.14.3), and **SlideViewInfoContainer** (section 2.4.21.9).

PowerPoint 2002, Office PowerPoint 2003, Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 additionally preserve unknown records found in the following records: **ClientVisualElementContainer** (section 2.8.44), **ExtTimeNodeContainer** (section 2.8.15), **TimeAnimateBehaviorContainer** (section 2.8.29), **TimeAnimationValueListContainer** (section 2.8.31), **TimeBehaviorContainer** (section 2.8.34), **TimeColorBehaviorContainer** (section 2.8.52), **TimeCommandBehaviorContainer** (section 2.8.71), **TimeConditionContainer** (section 2.8.75), **TimeEffectBehaviorContainer** (section 2.8.61), **TimeMotionBehaviorContainer** (section 2.8.63), **TimePropertyList4TimeNodeContainer** (section 2.8.18), **TimeRotationBehaviorContainer** (section 2.8.65), **TimeScaleBehaviorContainer** (section 2.8.67), **TimeSetBehaviorContainer** (section 2.8.69), and **TimeStringListContainer** (section 2.8.36).

<2> Section 2.1.5: PowerPoint 2002, Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 create this stream for encrypted documents, but the contents of the stream will be empty.

<3> Section 2.1.7: PowerPoint 97 and PowerPoint 2000 always omit and ignore this storage, because these versions do not support signatures. PowerPoint 2002 and Office PowerPoint 2003 also omit and ignore this storage and instead use the "_signatures" stream.

<4> Section 2.1.9: PowerPoint 97 and PowerPoint 2000 always omit and ignore this stream, because these versions do not support signatures. PowerPoint 2002 and Office PowerPoint 2003 do not omit this stream.

<5> Section 2.2.3: The object models in PowerPoint 97, PowerPoint 2000, PowerPoint 2002, and Office PowerPoint 2003 allow bullet sizes to be specified as percentages outside the range of 25 to 400, inclusive.

<6> Section 2.3.2: PowerPoint 2002 uses the **headerToken** 0xE391C05F for encrypted documents.

<7> Section 2.3.2: PowerPoint 2000 initially used the value 0x00000009 to indicate to PowerPoint 97 that a file contained multiple main master slides and therefore could not be opened by PowerPoint 97. PowerPoint 97 was subsequently updated to add support for reading files with multiple main master slides, and PowerPoint 2000 was later updated to stop writing 0x00000009 to indicate the presence of multiple master slides.

<8> Section 2.3.2: PowerPoint 97 omits this field.

<9> Section 2.3.3: PowerPoint 97, PowerPoint 2000, PowerPoint 2002, Office PowerPoint 2003, Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 write the minor build version of powerpnt.exe to this field.

<10> Section 2.3.3: PowerPoint 97 and PowerPoint 2000 will always omit this field because they do not support opening or creating encrypted documents.

<11> Section 2.3.7: PowerPoint 2002 writes 0xE391C05F to the **headerToken** field for an encrypted document.

<12> Section 2.4.15.2: PowerPoint 97 behaves unpredictably if **formatId** is 0x000D; PowerPoint 2000, PowerPoint 2002, Office PowerPoint 2003, Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 interpret it as 0x0000 instead.

<13> Section 2.4.16.5: PowerPoint 97, PowerPoint 2000, PowerPoint 2002, Office PowerPoint 2003, Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 store the file extension of the source sound file initially embedded. Regardless of the file extension, only WAV and AIFF audio files are supported by these versions for playback.

<14> Section 2.4.17.1: Only PowerPoint 2000 fully supports the **BroadcastDocInfo9Container** record. PowerPoint 2002 supports a subset of this record necessary for archiving a presentation broadcast.

<15> Section 2.4.23.1: PowerPoint 97 sets this field to 0x001.

<16> Section 2.4.23.4: PowerPoint 97 ignores this record.

<17> Section 2.4.23.4: PowerPoint 97 and PowerPoint 2000 ignore this record.

<18> Section 2.4.23.4: PowerPoint 97, PowerPoint 2000 and PowerPoint 2002 ignore this record.

<19> Section 2.4.23.4: PowerPoint 97, PowerPoint 2000, PowerPoint 2002, and Office PowerPoint 2003 ignore this record.

<20> Section 2.4.23.5: PowerPoint 2000 and PowerPoint 2002 do not ignore this field.

<21> Section 2.4.23.5: PowerPoint 2000 does not ignore this field.

<22> Section 2.4.23.5: PowerPoint 2000 does not ignore this field.

<23> Section 2.4.23.5: PowerPoint 2000 and PowerPoint 2002 do not ignore this field.

<24> Section 2.4.23.6: PowerPoint 2002 and Office PowerPoint 2003 do not ignore this field.

<25> Section 2.4.23.6: PowerPoint 2002 and Office PowerPoint 2003 do not omit this field when the document contains embedded reviewer documents.

<26> Section 2.4.23.6: PowerPoint 2002 and Office PowerPoint 2003 do not ignore this field.

<27> Section 2.4.23.6: PowerPoint 2002 and Office PowerPoint 2003 do not omit this field when the document contains embedded reviewer documents.

<28> Section 2.4.23.6: PowerPoint 2002 and Office PowerPoint 2003 do not ignore this field.

<29> Section 2.4.23.6: PowerPoint 2002 and Office PowerPoint 2003 do not omit this field when the document contains embedded reviewer documents.

<30> Section 2.4.23.8: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not omit this field.

<31> Section 2.5.1: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not preserve this field.

<32> Section 2.5.1: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this field.

<33> Section 2.5.1: PowerPoint 97 does not preserve this field.

<34> Section 2.5.1: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not preserve this record.

<35> Section 2.5.2: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<36> Section 2.5.2: PowerPoint 97 does not preserve this record.

<37> Section 2.5.2: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<38> Section 2.5.2: PowerPoint 97 does not preserve this record.

<39> Section 2.5.2: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<40> Section 2.5.2: PowerPoint 97 does not preserve this record.

<41> Section 2.5.2: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<42> Section 2.5.2: PowerPoint 97 does not preserve this record.

<43> Section 2.5.2: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<44> Section 2.5.2: PowerPoint 97 does not preserve this record.

<45> Section 2.5.2: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<46> Section 2.5.2: PowerPoint 97 does not preserve this record.

<47> Section 2.5.2: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<48> Section 2.5.2: PowerPoint 97 does not preserve this record.

<49> Section 2.5.3: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this field.

<50> Section 2.5.3: PowerPoint 97 does not preserve this field.

<51> Section 2.5.3: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not preserve this field.

<52> Section 2.5.3: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not preserve this field.

<53> Section 2.5.4: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<54> Section 2.5.4: PowerPoint 97 does not preserve this record.

<55> Section 2.5.4: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<56> Section 2.5.4: PowerPoint 97 does not preserve this record.

<57> Section 2.5.4: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<58> Section 2.5.4: PowerPoint 97 does not preserve this record.

<59> Section 2.5.4: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<60> Section 2.5.4: PowerPoint 97 does not preserve this record.

<61> Section 2.5.4: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<62> Section 2.5.4: PowerPoint 97 does not preserve this record.

<63> Section 2.5.4: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<64> Section 2.5.4: PowerPoint 97 does not preserve this record.

<65> Section 2.5.4: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<66> Section 2.5.4: PowerPoint 97 does not preserve this record.

<67> Section 2.5.4: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<68> Section 2.5.4: PowerPoint 97 does not preserve this record.

<69> Section 2.5.6: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not preserve this record.

<70> Section 2.5.7: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<71> Section 2.5.7: PowerPoint 97 does not preserve this record.

<72> Section 2.5.7: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<73> Section 2.5.7: PowerPoint 97 does not preserve this record.

<74> Section 2.5.7: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<75> Section 2.5.7: PowerPoint 97 does not preserve this record.

<76> Section 2.5.8: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not preserve this record.

<77> Section 2.5.9: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<78> Section 2.5.9: PowerPoint 97 does not preserve this record.

<79> Section 2.5.9: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<80> Section 2.5.9: PowerPoint 97 does not preserve this record.

<81> Section 2.5.22: PowerPoint 97 ignores this record.

<82> Section 2.5.22: PowerPoint 97 and PowerPoint 2000 ignore this record.

<83> Section 2.5.22: PowerPoint 97, PowerPoint 2000, PowerPoint 2002, and Office PowerPoint 2003 ignore this record.

<84> Section 2.5.24: PowerPoint 2002 and Office PowerPoint 2003 do not ignore this field.

<85> Section 2.5.24: PowerPoint 2002 and Office PowerPoint 2003 do not omit this field when the document contains embedded reviewer documents.

<86> Section 2.5.24: PowerPoint 2002 and Office PowerPoint 2003 do not ignore this field.

<87> Section 2.5.24: PowerPoint 2002 and Office PowerPoint 2003 do not omit this field when the document contains embedded reviewer documents.

<88> Section 2.7.4: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<89> Section 2.7.4: PowerPoint 97 does not preserve this record.

<90> Section 2.7.4: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<91> Section 2.7.4: PowerPoint 97 does not preserve this record.

<92> Section 2.7.4: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<93> Section 2.7.4: PowerPoint 97 does not preserve this record.

<94> Section 2.7.4: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this record.

<95> Section 2.7.4: PowerPoint 97 does not preserve this record.

<96> Section 2.7.6: PowerPoint 97 and PowerPoint 2000 ignore this field.

<97> Section 2.7.9: PowerPoint 97, PowerPoint 2000, PowerPoint 2002 and Office PowerPoint 2003 do not ignore this flag.

<98> Section 2.7.9: PowerPoint 97, PowerPoint 2000, PowerPoint 2002 and Office PowerPoint 2003 do not ignore this flag.

<99> Section 2.7.14: PowerPoint 97 sets this field to 0x001 or 0x002.

<100> Section 2.7.17: PowerPoint 97 ignores this record.

<101> Section 2.7.17: PowerPoint 97 and PowerPoint 2000 ignore this record.

<102> Section 2.7.17: PowerPoint 97, PowerPoint 2000 and PowerPoint 2002 ignore this record.

<103> Section 2.8.1: PowerPoint 97 and PowerPoint 2000 do not ignore this field. PowerPoint 2002, Office PowerPoint 2003, Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not ignore this field if an animation hash computed as specified in the HashCode2.8.3Atom record does not match the value stored in the HashCode2.8.3Atom record itself. If the two hashes do match, the contents of this field are ignored in favor of the contents of the **ExtTimeNodeContainer** record contained by the **SlideContainer** (section 2.5.1) or **MainMasterContainer** (section 2.5.3) record that contains this **AnimationInfoContainer** record. If the two hashes do not match, the contents of the **ExtTimeNodeContainer** record are instead ignored in favor of the contents of the **AnimationInfoAtom** records.

<104> Section 2.8.2: If this field is -1, PowerPoint 97 and PowerPoint 2000 play the animation for this shape before all other shapes; PowerPoint 2002, Office PowerPoint 2003, Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 ignore the animation for the shape.

<105> Section 2.8.2: PowerPoint 2002 and Office PowerPoint 2003 always write out 0x00.

<106> Section 2.8.10: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 set this value as less than or equal to 0x00000009.

<107> Section 2.9.53: PowerPoint 97, PowerPoint 2000, PowerPoint 2002, Office PowerPoint 2003, Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 ignore this record when **end** – **begin** is greater than or equal to 256.

<108> Section 2.10.28: PowerPoint 2000 writes other positive values besides 0x01. All values not equal to 0x00 are interpreted as 0x01.

<109> Section 2.10.28: PowerPoint 97 does not ignore this field.

<110> Section 2.11.13: Prior to SP1, Office PowerPoint 2007 set this field to 0xF.

<111> Section 2.11.21: PowerPoint 97 does not preserve this record.

<112> Section 2.11.21: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 do not omit this record.

<113> Section 2.13.11: Office PowerPoint 2007, PowerPoint 2010, and PowerPoint 2013 use ExOleSub_Visio; PowerPoint 97, PowerPoint 2000, PowerPoint 2002, Office PowerPoint 2003 use **ExOleSub_Default** for **ProgID Visio.Drawing**.

<114> Section 2.13.11: Office PowerPoint 2007 starting with SP2, PowerPoint 2010, and PowerPoint 2013 use **ExOleSub_WordODF**. PowerPoint 97, PowerPoint 2000, PowerPoint 2002, Office PowerPoint 2003 use **ExOleSub_Default** for **ProgID Word.OpenDocumentText**.

<115> Section 2.13.11: Office PowerPoint 2007 starting with SP2, PowerPoint 2010, and PowerPoint 2013 use **ExOleSub_ExcelODF**. PowerPoint 97, PowerPoint 2000, PowerPoint 2002, Office PowerPoint 2003 use **ExOleSub_Default** for **ProgID Excel.OpenDocumentSpreadsheet**.

<116> Section 2.13.11: Office PowerPoint 2007 starting with SP2, PowerPoint 2010, and PowerPoint 2013 use **ExOleSub_PPTODF**. PowerPoint 97, PowerPoint 2000, PowerPoint 2002, Office PowerPoint 2003 use **ExOleSub_Default** for **ProgID PowerPoint.OpenDocumentPresentation**.

6 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

7 Index

A

B

C

D

E

F

G

H

I

K

L

M

N

O

P

R

S

T

U

V

W

Z