Dynamics in Search User Interfaces

Marcus Nitsche, Florian Uhde, Stefan Haun and Andreas Nürnberger
Otto von Guericke University, Magdeburg, Germany
{marcus.nitsche, stefan.haun, andreas.nuernberger}@ovgu.de,
florian.uhde@st.ovgu.de

ABSTRACT

Searching the WWW has become an important task in today's information society. Nevertheless, users will mostly find static search user interfaces (SUIs) with results being only calculated and shown after the user triggers a button. This procedure is against the idea of flow and dynamic development of a natural search process. The main difficulty of good SUI design is to solve the conflict between good usability and presentation of relevant information. Serving a UI for every task and every user group is especially hard because of varying requirements. *Dynamic search user interface elements* allow the user to manage desired information fluently. They offer the possibility to add individual meta information, like tags, to the search process and enrich it thereby.

Keywords

Search User Interface, User Experience, Exploratory Search.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval.; H.5.2 [**Information Interfaces and Presentation**]: User Interfaces.

General Terms

Design, Human Factors, Management.

1. MOTIVATION

Since the launch of the WWW, users accumulated a vast amount of information. With broadband technologies becoming a part of everyday life[1] the WWW offers a great opportunity in terms of learning and education. University courses, for instance, are available online and nearly every topic is handled somewhere in the great amount of blogs, Q&A pages, fora, web pages or databases. Yet there is no map, no guide leading through this vast amount of information. Users need to search for information, to locate the bits fitting to their specific information need, indexing the amount of knowledge available online. Therefore, a proficient tool to analyse the structure of the web and to provide guidance to specific sources of information is needed. This task is accomplished by modern search engines like *Google*[2], *Bing*[3], *Yahoo*[4] and other local or topic centred search engines. By the increase of computational power in smart phones and wider access to online resources the demand for these search tools has risen and the quality of the search terms has changed. Instead of single-query-searches, users tend to request complex answers[5], trying to learn about topics in deep. While the need for information and the expectations of users increased, matching the broader knowledge base contained in the Internet in the last few years. About 300 Mio. websites were added in 2011[6]. Search engines mainly remain the same. This leads to the fact that a *"significant design challenge for web search engine developers is to develop functionality that accommodates the wide variety of skills and information needs of a diverse user population"* [1]. Therefore, this paper proposes the concept of using *dynamic elements* in SUIs, that focus on fluent work flow characteristics, a high grade of interactivity and an adequate answer-time-behaviour.

2. INFORMATION GATHERING

Looking at users' habits in search, they no longer perform simple lookup searches. There is an increasing need to answer complex information needs. Therefore, we mainly consider information gathering processes, searches where users are not familiar with the domain. Users need to refine search queries, branch out into other queries to gain additional understanding and collect results to merge them into a single topic. This kind of search process is called *exploratory search* and is contrary to a *known-item search* task as stated in [2]. Exploratory search processes *"depend on selection, navigation, and trial-and-error tactics, which in turn facilitate increasing expectations to use the Web as a source for learning and exploratory discovery"* [3]. Search tasks are fragmented, consisting of single queries and search requests. The search requests may yield additional data or parts of the final information which in the end form the information requested by the user. While performing such a complex search task, a pattern called *berry picking* [4] can be observed. While reading through a source of data, looking for qualified information the user discovers new *traces* leading to other sources, which have to be handled one after the next. By re-

[1] `http://www.internetworldstats.com/images/world2012pr.gif`, 02.05.2013

[2] `http://www.google.com`, 02.05.2013

[3] `http://www.bing.com`, 02.05.2013

[4] `http://www.yahoo.com`, 02.05.2013

[5] see the 2009 HitWise study for more details: `http://image.exct.net/lib/fefc1774726706/d/1/SearchEngines_Jan09.pdf`, 10.07.2013

[6] `http://royal.pingdom.com/2012/01/17/internet-2011-in-numbers/`, 02.05.2013

fining the search and gaining deeper information the user satisfies the initial need for it. These different traces span a map in the end, representing the whole search and its processing. When someone is learning about something this map is refined and expanded. The learner may track back to a certain node and deepen the understanding about it by adding new queries, and therefore new branches. Or he may discard a whole part of the map because it turned out that the contained information was not relevant to him. When the user is satisfied with the gained information this map is encapsulated and represents the whole development of this complex information. According to this concept the result is not a single object. It is a set of sources, representing the learning process for a specific user.

Looking at the current process of information gathering in the Internet there are only two *places*. The Internet itself, containing the pool of existing information, in an unstructured form and a mental model about the information (space) that is constructed. This system may work perfect when dealing with short, exact search queries like *postal code New York City*, but when it comes to complex information needs, where the user needs to access a lot of information and generate more detailed search queries while looming through pages this system reaches it boundaries. The user might retrieve only partial facts. For example, if the user needs explanation of a term used in its initial query. The user is now in need of another place, where he can store information, reorder it and put it into the context of other information pieces.

3. STATE OF THE ART

Looking at *Google*, the most used search engine today [5], the user interface of a modern search engine is mostly static. *Google's* features include some dynamic elements like real time search. For example *"[..] Google Suggest which interactively displays suggestions in a drop-down list as the searcher types in each character of his/her query. The suggestions are based on similar queries submitted by other users."* [1] Dynamic previews of results will be offered when clicking on the double arrow beside a result. But the core of the interface has not changed a lot since its launch in 1997[7]. While adopting fast to new information sources like Facebook and Twitter, Google discarded the adoption of new HCI methods in favour of a clean, slim interface. With increasing touch support on the devices, a richer user interface can be designed to provide the user with immediate feedback and allows haptic interaction with the search process. Some mobile clients take advance of the additional information available, like the iOS search client, which switches to voice queries when the phone is lifted to the head, but there is no full extension of *Google's* search services. While *Google* is an adequate tool for short queries and queries calling for a direct answer, features for deep research on complex topics are missing.

One way to integrate *dynamic elements* into existing SUI infrastructure is to build an overlay. Thereby, dynamic UI utilize existing, well known search engines and provide a benefit by enriching them. This approach is shown in the *Boolify*[8] search engine, which provides a dynamic drag and drop interface on top of *Google's* search engine. This engine is relatively new and was build to promote the understanding of boolean queries. Users *build* a query by dragging jigsaw like parts onto a search surface. These parts contain words (general or exact) and linkers like AND and OR. Additional parts have been added to provide search on a specific page or for synonyms. By adding and linking those parts the user constructs a boolean query which will be submitted to the *Google* search engine. *Boolify* was built for children and elderly. Tests in a third grade technology class showed that children without any knowledge of boolean queries were able to construct complex queries just by pulling them together piece by piece[9]. A similar approach was implemented at *SortFix*[10]. This tool offers the user the *"ability to drag and drop search terms in between several* buckets*"* [6] to in- and exclude them in the query. With a *Standy Bucket* users are *"able to keep track of all [their] inspirations and alternative search words off to the side, ready to be dragged and dropped into your search box if needed."* [6] Another possible use of dynamic interface elements is the weighting of search terms based on their font size as used at *SearchCloud.net*[11]. The ranked keywords are shown in a Tag Cloud like manner and additionally the site shows, based on the ranking, *"the calculated relevance score for each [result]"* [6]. Not only the query building process can be enchanted by dynamic elements, also the presentation of the result can benefit from it. Dynamic side loading can provide the user a lens like view to parts of the result where keywords occur. *Microsoft's WaveLens "[...] fetches a longer sample for the page containing your keywords, without you having to download it."* [8] *Microsoft Research* shows that in a study using WaveLens, presenting the participants with a normal interface and two versions of *WaveLens'* UI (instant zoom and dynamic zoom), *"participants were not only slower with the normal view than the other two, but they were more than twice as likely to give up"* [9]. Another way of result presentation was shown at *SearchMe*[12]: *"Fragmentation into multiple sites, domains and identities becomes a huge distraction. User don't know which site to visit for which purpose, and the lack of consistent, intuitive inter-site search and navigation makes it hard to find content [..]"* [6]. All these dynamic features can be used as a mask over traditional SUIs to extend them. By hiding the dynamic part, dynamic elements can be added to an existing search engine and let the user make a choice which part should be shown and used. The proposed concept is similar to Byström & Hansen's approach in [19].

Issues. Comparing the state of the art with the process of information gathering some issues appear, which may be resolved or at least damped by using of dynamic elements. While collecting information pieces for solving complex questions the user discovers new sources, containing more information. These sources may not form a linear search process every time. Sometimes there will be a split and the user needs to decide which trace to follow first. This issue is also noted in [10]. Today's search engines offer only little support for this. The user needs to save web pages to favourites or organize them himself for later reading. Searching different terms one by one allows users to follow new pages like traces through the Internet. By connecting these traces and setting them into relation the user can retrieve the whole information needed to cover his query. Most modern search engines discard this feature, it is again something the user needs to do by himself. This leads to another more general problem, the *enclosing of search queries*. *Google* for example handles every search term as a new operation. Data is stored, but contains only general information about the user, queries are not related to each other and therefore miss-

[7] `http://www.google.com/about/company/history/`, 02.05.2013
[8] `http://www.boolify.org/`, 02.05.2013
[9] `http://ed-tech-axis.blogspot.de/2009/03/boolified.htm`, 02.05.2013
[10] *SortFix.com*, offline since 11/2011, Firefox plugin: `https://addons.mozilla.org/en-us/firefox/addon/sortfix-Extension`, 02.05.2013
[11] `http://searchcloud.net/`, 02.05.2013
[12] `http://www.searchme.com`, offline since 2009

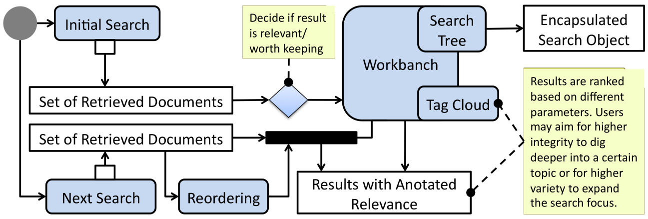

Figure 1: Data flow while refining during search.

ing its broader context. But when learning about a complex topic refining the search query is more important to the user. In the iteration of search processes, to narrow down the mass of information and to tap new sources, the searcher needs to rewrite and modify the query, to link it to other related search tasks. Building a connection between parts of information and evaluating it against each other is a core principle of learning. This leaves the user targeting a broader, intense search, in the need to build a custom solution to extract knowledge and manage it. This is strictly against the guideline for online interfaces which suggests to *"[..] not require users to remember information from place to place on a Web site"* [11] as this is a distraction from the main process of searching and destroys the interaction flow triggered by the search process.

4. COMPOSING A DYNAMIC SUI

The proposed approach shows a design based on today's search engines, enriched with dynamic UI elements to provide a plus for the user. The design includes principles to form web based learning applications [12] to focus on the completion of complex search tasks. By adding dynamic elements internal states can be visualized for the user to give a better overview about the current position in the search process. Furthermore it will allow the serialization of search processes and to step in at every point of the process later on. As stated in Beyond Box Search *"different interfaces (or at least different forms of interaction) should be available to match different search goals"* and *"[t]he interface should facilitate the selection of appropriate context for the search"* [13]. Both of this quality measurements should be regarded when conceptualizing a SUI. The first point will be covered by a modular UI, the user may move, hide and scale elements to fit his current need. The second point is strongly bounded to the use of dynamic items in the UI design. By giving immediate feedback to the user it is easier to classify the current results. The context of the whole search process will be persistent over multiple search queries and provide a method of accumulation parts of the search process into a single object.

Four *features* are proposed and explained in this paper, showing a use-case for dynamic search interfaces and giving a suggestion how this can be accomplished. Together these features build up a mid instance to accumulate into a bigger context for a search process. This clipboard (Fig. 1) reshapes the search process and provide the place to store information between search queries. Instead of trying to accumulate knowledge and information directly the user is able to construct a solution of the search query in this buffer and save it as a complete collection of the information retrieval process.

Reordering. Giving users the opportunity to reorder and therefore to rate a search result is an important step towards dynamics in SUIs. Every result is handled as a single item and can be picked by the user and dropped in another place. The other items reorder fluently, giving user feedback while the user moves on. The SUI holds an array of parameters, which is used to evaluate every item. Possible criteria are *Accuracy*, *Clarity*, *Currency* and *Source Novelty*. These and more criteria are mentioned and explained in [14]. When a user reorders items to fit his preferences the search engine may use the information provided by this ranking to weight the existing parameters to yield better results in the future. The engine will be able to present results ranked according to the user's preference. This can be done for all users and also search process wide, as some search tasks require documents and papers while others may focus on web pages or media. This addition to classical user interfaces can make great use of the up-trend for touch based devices, in 2012 89% of mobile phones and smart-books support touch [15]. Designing the SUI responsive to touch and gesture is maybe one of the most natural solutions for human computer interaction and adds an amount of possible actions based on gestures.

Workbench. The workbench targets the issue of loosing information while switching between different searches. It adds a third place to the proposed search process, located outside of the search scope but still related to it. The user may drop queries here to keep them throughout the whole search process. When entering a query, indicators show how relevant items on the bench are. This allows the user to classify new results in terms of integrity towards already selected snippets. The workbench acts as a buffer between search queries, adding a broader context to every entry. Like a frame, it contains information exclusively attached to the current search process, leading to the possibility of customization and user centred search environments. When the user switches between queries he can immediately determine how well the new results fit into already selected items. This allows identifying false positive as well as exploratory search [16] results. Users may just enter queries that lead to a peripheral topic and check the indicators whether the result is relevant to his initial information.

Tag Cloud. The tag cloud is another feature to guide the user in the search process. As shown in [17] a tag cloud supported retrieval system can increase the find rate of adjacent data nodes by nearly 15%. When adding an item to the workbench its most relevant tags are extracted and visualized in the tag cloud. It is able to show how often a tag occurs and how different tags are related to each other. When entering a new search query the tag cloud displays the relevant tags and reorders the cloud to revolve around the current tags. By combining distance and size of the entered tag with their direct neighbours the user can directly spot how homogeneous its current query is in terms of the whole process. The tag cloud can also use the existing tags to show the user other closely related tags and suggest query refinement based on tag proximity. Colours can indicate the state a tag is currently in. A possible color scheme for western culture can be based on the three colors used in traffic lights. The concept of three-coloured traffic lights also work for color-blind people, since they do have a given position. Therefore, we also use second coding paradigm: form. A green triangle is proposed for tags resulting from the current query, which are contained in the overall tag cloud spanned by the workbench. An orange circle indicates a warning for tags, either in the current query result or the bench, which are not related to the rest of the cloud. A red square is avoided for the reason that uncontained tags may not be bad, they can lead to a new direction or add a reasonable value to the whole search process. The tags are scaled depending on their frequency. When the user selects any item from the bench or the search result the corresponding tags are centred. The other tags are located based on their coherence with the selected tags; closer means the tag is in a direct relation to the selected item. A user can quickly

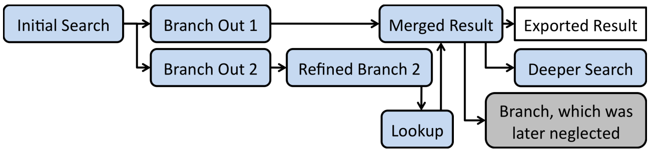

Figure 2: Search map, representing the search process.

check the integrity of his search process by looking at the tag cloud. A slim, packed cloud means the results are all related to each other, an open, wide cloud indicates a broad result field, covering many aspects. False positives may be filtered out, when enough items exist, as they stick out the rest of the cloud.

Search Map Support. The search map (Fig. 2) acts as a representation of the whole search process, by storing every query and following up querying and visualize it in a chronological order. The user may select single nodes in the map to get into the state of search process at this moment and refine it. The map provides a kind of top view to the path of the search and shows where the user branched out into new queries. It allows the user to cut off nodes and whole branches if they are not needed any more to fulfil the need for information. As it contains every action and some data in the current search process, the search map might be serialized and stored to retrieve the search process later on. With this map at hand a user can save whole search tasks just like he saves favourite web pages. He can step back into the process at any time and reconstruct the whole learning process or correct parts of the search which has proven to be not correct. This kind of *Story Telling* helps to visualize the given data, *"[...] lead to findings, which prompt actions [...] [and] can indicate the need to forage for new data."* [18] The search map [7] features two ways of expanding. The user may follow a result to expand it vertically. The result is added as a new node and resides in the map until it is processed further. When the user selects an existing node he steps back to the vertical position of this node and can now branch out horizontally. This deals with an issue of berry-picking [4], where the new sources has to be processed one by one. While not abolishing this the search map provides a visual representation to simulate parallelism. The map also allows scoping of the analysis by creating a horizontal or vertical bound. Only tags and items inside this bound will be considered, the rest is greyed out. This allows the user to dig deep into a certain topic (small vertical bounds) or create a better understanding of a certain term and add more results to a certain query (horizontal boundary). This can help the user to concentrate on smaller pieces of a big search process and to narrow down problems one by one.

5. CONCLUSION

This paper has shown certain design flaws of today's search engines and some proposed dynamic design principles to counter them. The application of the envisioned elements can extend a search engine towards a software capable of complex research tasks. With the current up-trend of online learning this unlock a new way of using them. The surplus resides not only in the dynamic and vivid interface, it prepares a whole new tier of online search solutions. The process of learning can be preserved and shared with others. One can come back at any time, jump right into the saved search process and reconstruct the development of certain knowledge. With this tool chain at hand learning becomes a social and an integrative part of the WWW. The next step in deploying dynamic elements into search user interfaces would be prototyping them. Design snippets need to be tested for usability and acceptance in the real world. Starting as overlays and additional feature of existing search engines may develop and emerge into independent solutions.

Acknowledgement

Part of the work is funded by the German Ministry of Education and Science (BMBF) within the ViERforES II project (01IM10002B).

6. REFERENCES

[1] Sandvig, J. C., Deepinder B.: User Perceptions of Search Enhancements in Web Search. In: J. of Comp. Inform. Syst. 52, no. 2, 2011.

[2] White, R. W., Marchionini, G.: A Study of Real-Time Query Expansion Effectiveness. In: SIGIR Forum 39, 2006.

[3] Marchionini, G.: Exploratory Search: From Finding to Understanding. In: Comm. of the ACM 49, 4.2006.

[4] Bates, Marcia J.: The design of browsing and berrypicking techniques for the online search interface. Univers. of Calif. at L.A., 1989.

[5] Purcell, K., Brenner, J., Rainie, L.: Search Engine Use 2012. In: Pew Internet & American Life Project, 2013.

[6] Bates, M. E.: Make Mine Interactive. Vol. 31, Issue 10, p. 63, 12/2008.

[7] Heer, J., Viégas, F. B., Wattenberg, M.: Voyagers and voyeurs: Supporting asynchronous collaborative visualization. In: Commun. of the ACM, 52, No. 1, pp. 87–97, ACM, New York, NY, USA, 01/2009.

[8] MS Research: Cutting Edge. New Scientist 181, no. 2434, 2004.

[9] Paek, T., Dumais, S., Logan, R.: WaveLens: A new view onto Internet search results. In: Proc. of the SIGCHI Conference on Human Factors in Computing Systems (CHI '04), pp. 727–734, 2004.

[10] Morville, P., Callender, J.: Search Patterns - Design for Discovery. In: O'Reilly, 2010.

[11] U.S. Department of Health and Human Services, Research-Based Web Design and Usability Guidelines. Washington, D.C.: GPO, n.d.

[12] Jayasimman, L., Nisha Jebaseeli, A., Prakashraj, E.G., Charles, J.: Dynamic User Interface Based on Cognitive Approach in Web Based Learning. In: Int. J. of CS Iss. (IJCSI), 2011.

[13] Buck, S., Nicholas, J.: Beyond the search box. Reference & User Services 51(3), pp. 235-245, 2012.

[14] Beresi, U. C., Kim, Y., Song, D., Ruthven, I.: Why did you pick that? Visualising relevance criteria in exploratory search. In: Int. J. on Dig. Lib. 11 (2), pp. 59–74, 2010.

[15] Lee, D: The State of the Touch-Screen Panel Market in 2011. In: Walker Mobile, LLC, SID Information Display Magazine, 3.2011.

[16] White, R. W., Kules, B., Drucker, S. M., schraefel, m. c.: Supporting Exploratory Search. In: Comm. of the ACM 49, 4.2006.

[17] Trattner, C.: QUERYCLOUD: Automatically linking related documents via search query (Tags) Clouds. In: Proc. of the IADIS Int. Conf. on WWW/Internet, 2010.

[18] Mackinlay, J. D.: Technical Perspective: Finding and Telling Stories with Data. In: Comm. of the ACM 52, 2009.

[19] Byström, K., Hansen, P.: Conceptual framework for tasks in information studies: Book Reviews. In: J. Am. Soc. Inf. Sci. Technol., Vol. 56, 10, pp. 1050–1061, John Wiley & Sons, Inc., New York, NY, USA, 2005.