Baselines & Benchmarks – Making Open Source Big Data Analytics Easy

Arjuna Chala, Senior Director
Special Projects for HPCC Systems®

Today’s Presenter

Arjuna Chala

Senior Director, Special Projects, HPCC Systems

arjuna.chala@lexisnexisrisk.com, @arjunachala

Arjuna Chala is Sr. Director of Special Projects for the HPCC Systems® platform at LexisNexis Risk Solutions®. With almost 20 years of experience in software design, Arjuna leads the development of next generation big data capabilities including creating tools around exploratory data analysis, data streaming and business intelligence. Arjuna strives to understanding new technologies and bring innovative applications and design to the HPCC System platform.

Dedicated to development excellence, Arjuna served as a key member of the team to bring the HPCC Systems platform to the open source community. In his work with HPCC Systems community leaders and system integrator partners, Arjuna’s efforts have contributed to the spread of HPCC Systems technology into the enterprise domestically as well as the international markets of China, Brazil, Europe and India.

Arjuna has a BS in Computer Science from RVCE, Bangalore University.

HPCC SYSTEMS®

Baselines and Benchmarks

Arjuna Chala

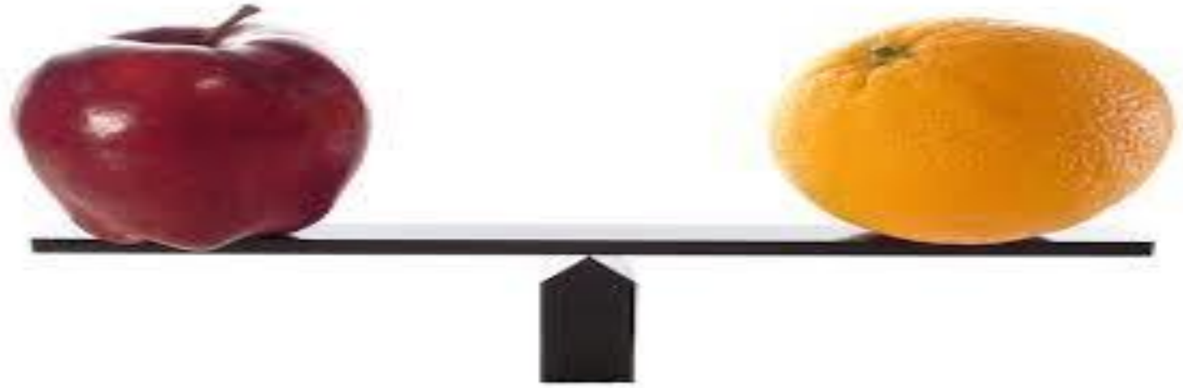
RELX Group

HPCC Systems Vs. Apache Spark

Are Benchmarks Important?

Answer: Yes, as an entry criteria

Baselines



Baseline 1 : HPCC Systems and Spark are solving similar Data Problems

Data Problems

- Not Clean – *e.g. Incorrect spelling, Incorrect Age Range*
- Misclassified Data – *e.g. Incorrect Gender Classification (Male vs. Female)*
- Missing Values – e.g. missing state or city in an address
- Has Duplication – e.g. repeating customer names
- High Noise Levels – e.g. data that is not relevant to the current problem
- Complex and Unstructured Data – e.g. financial information in a web page (html)
- Big Data - Terabytes
- Many Sources – CRM Systems, ERP Systems, Departments, Subsidiaries
- Complex Technology Solutions – Hadoop, Spark, HPCC Systems

Agree?

Well, we have a slightly different perspective

Problems

✓ Solutions

- ✓ Not Clean
- ✓ Misclassified Data
- ✓ Missing Values
- ✓ Has Duplication
- ✓ High Noise Levels
- ✓ Complex and Unstructured Data

- ✓ Many Sources
- ✓ Big Data
- ✓ Purpose Built Technology

HPCC SYSTEMS®

Examples

Data Profiling Anyone?

Previous polls have indicated that less than 10% of developers profile their data

Vehicle Telematics Data – Cologne Germany

Input

##	relative_timestamp	vehicle_id	x	y	speed
1	3627	59609802_17378	18862.5	26193.46	0
2	3631	59609802_17141	18862.5	26193.46	0
3	3635	559750	6533.07	12082.57	0
4	3636	559750	6532.395	12085.019	2.54
5	3637	559750	6531.21	12089.318	4.46
6	3638	559750	6529.563	12095.296	6.2

HPCC Data Profile

Pattern	Count
99.99	182,861,502
9.99	75,683,713
99.9	20,458,456
99999.99999999 9999	1

Indicates a speed >= 1000 meters/second

Summary Report

Record Count
354,358,979

There are a few records with blank y values

718,141 unique Vehicles

Recognized as a string. Expected to be a real8. Data is not clean

Column Report

Column	Input Type	Fill Rate	Fill Count	Cardinality	Output Type
relative timestamp	string	100	354,358,979	82387	unsigned3
vehicle id	string	100	354,358,979	718,141	string19
x	string	100	354,358,979	150,338,677	real8
y	string	99.999999	354,358,977	150,340,639	string32
speed	string	100	354,358,979	6993	string19

Is Deduplication Hard?

Previous polls have indicated that more than 60% of developers think it is NOT hard

Eliminate **FALSE NEGATIVES**

1. **Fl**avio Villanustre, Atlanta
2. **J**avio Villanustre, Atlanta

CORRECT

MATCH — the system has learnt that **"Villanustre"** **is specific** because the frequency of occurrence is **small** and there is **only one present in Atlanta**

ERROR

NO MATCH — because the rules determine that "Flavio" and "Javio" are not the same

INPUT

OUTPUT

Eliminate **FALSE POSITIVES**

1. **John Smith, Atlanta**
2. **John Smith, Atlanta**

ERROR

MATCH — because the rules determine that "John Smith" and the city for both the records match

CORRECT

NO MATCH — the system has learnt that **"John Smith"** **is not specific** because the frequency of occurrence is **large** and there are **many present in Atlanta**

Is a Single Source of Data Sufficient?

It depends. In most cases it is NOT

Record	First Name	Last Name	Address	City	State	Zip
1	MARCIA	MARSUPIAL	6035 JONES ST	ARVADA	CO	80004
2	KAREN	KANGAROO	5865 W OHIO AVE	LAKEWOOD	CO	80226

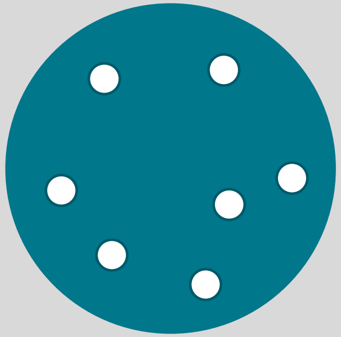
Are they the same person?

Match Source 1 @ time N	MARCIA	K	MARSUPIAL	6035		JONES	ST	ARVADA	CO	80004
Moved Source 1 @ time N +1	MARCIA	K	MARSUPIAL	9170	W	14TH	AVE	LAKEWOOD	CO	80226
Changed Last Name Source 2 @ time N +2	MARCIA	K	KANGAROO	9170	W	14TH	AVE	LAKEWOOD	CO	80226
Used Middle Initial Source 3 @ time N +2	K	MARCIA	KANGAROO	9170	W	14TH	AVE	LAKEWOOD	CO	80226
Match Source 1 @ time N + 3	KAREN	MARCIA	KANGAROO	5865	W	OHIO	AVE	LAKEWOOD	CO	80226

The Data Maturity Process

1: *COLLECT* Step

A single source of data is insufficient to overcome inaccuracies in the data



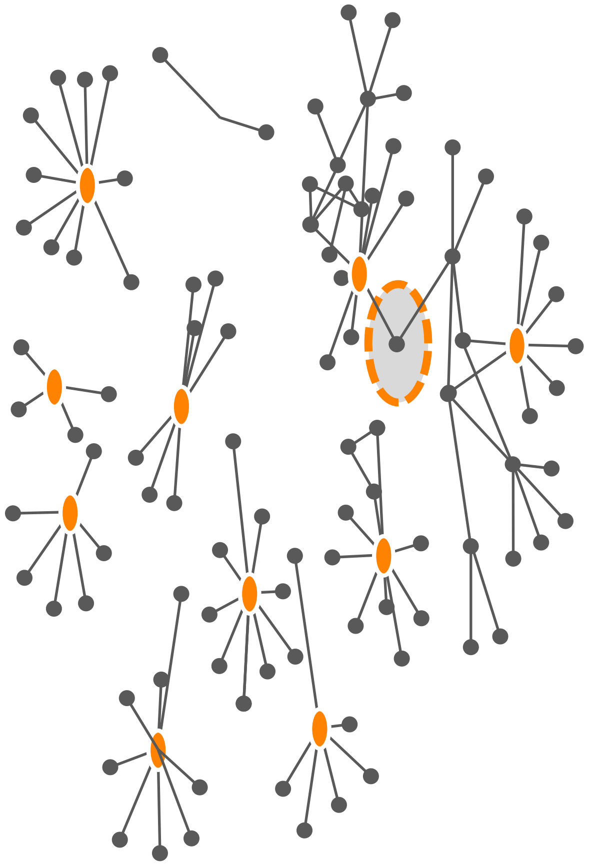
Our platform is built on the premise of absorbing data from **many data sources** and transforming them to **smart data (actionable)**

HPCC SYSTEMS®

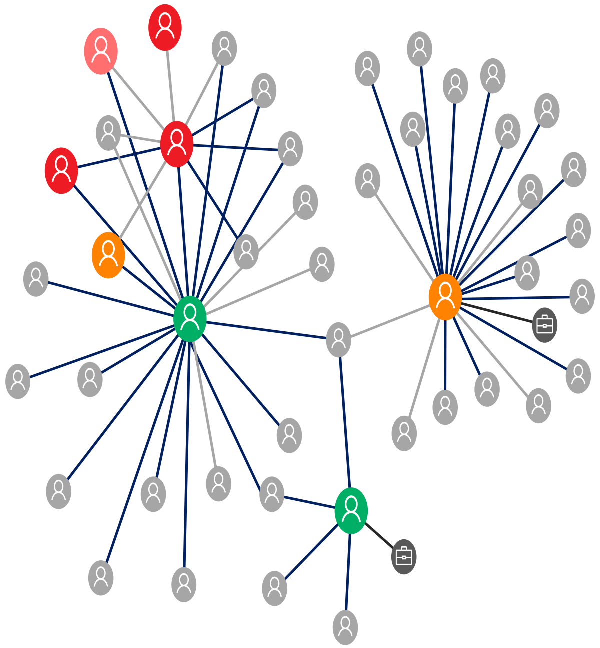
2: *LEARN* Step

- Raw Data
- Profile -> Clean -> Standardize
- Enhance -> Dedupe -> Relate
- Rich Connected Data

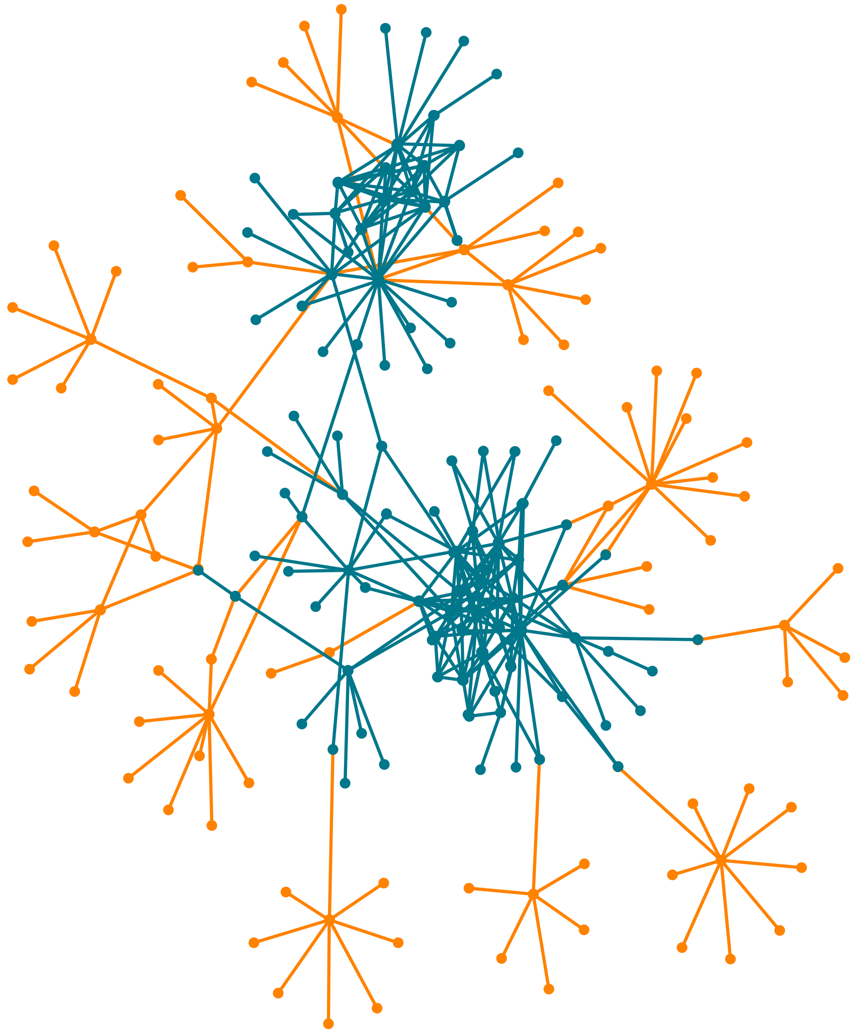
3: DECISION Step (Real-time or Batch)



INPUT: Disjointed Connections



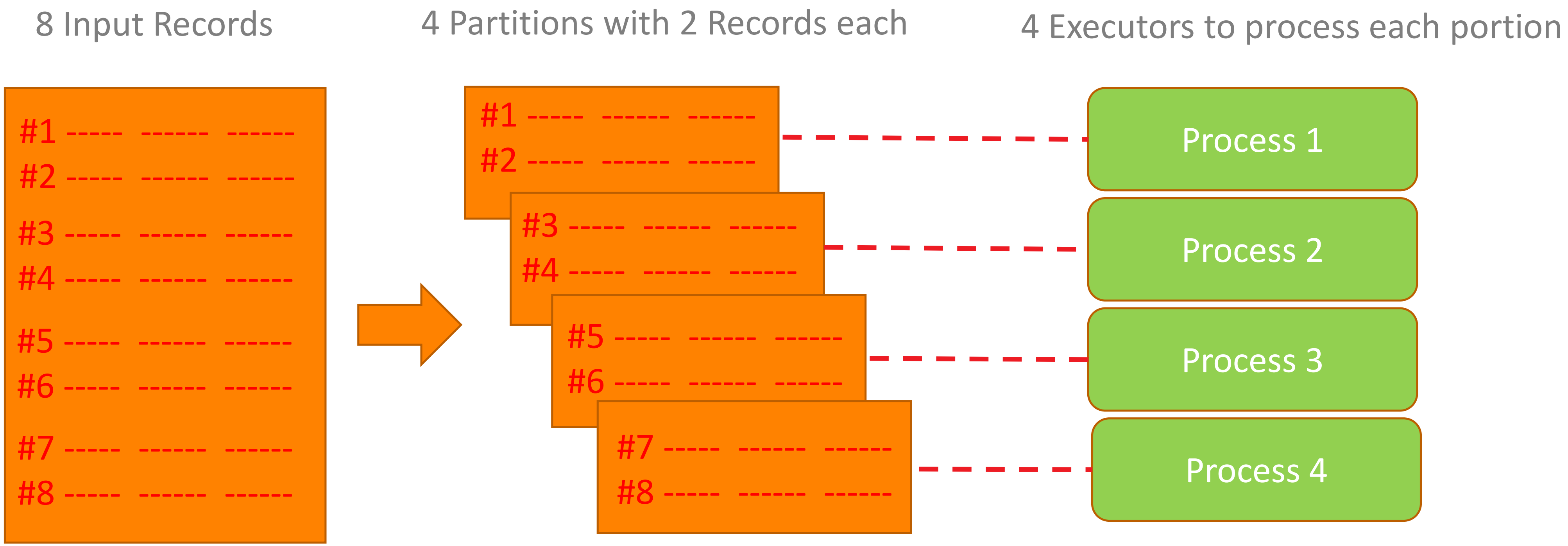
Query Graph



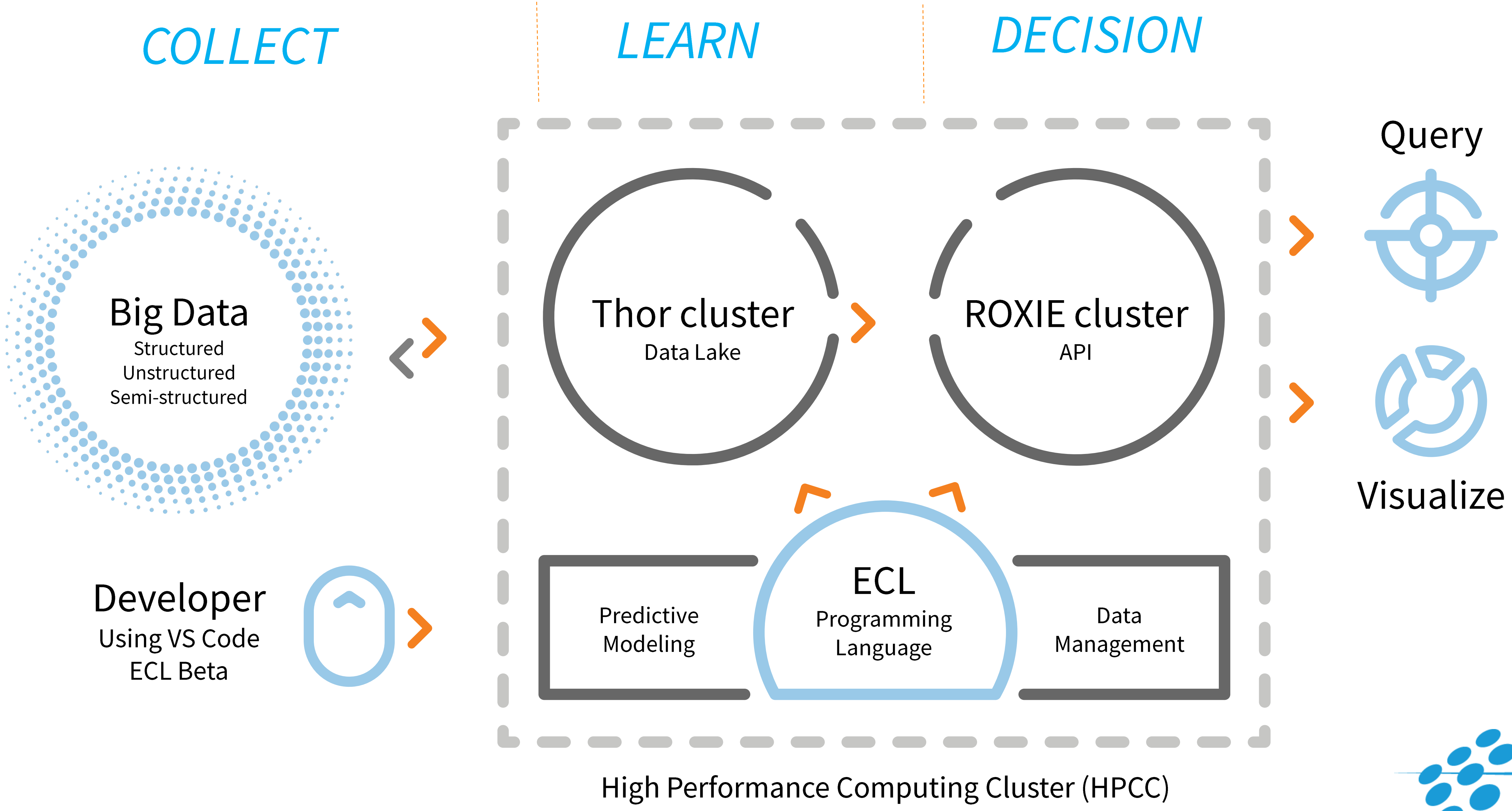
OUPUT: Rich Connections

Baseline 2 : HPCC Systems and Spark are similar in architecture

Data Partitions - Most data operations are inherently parallel

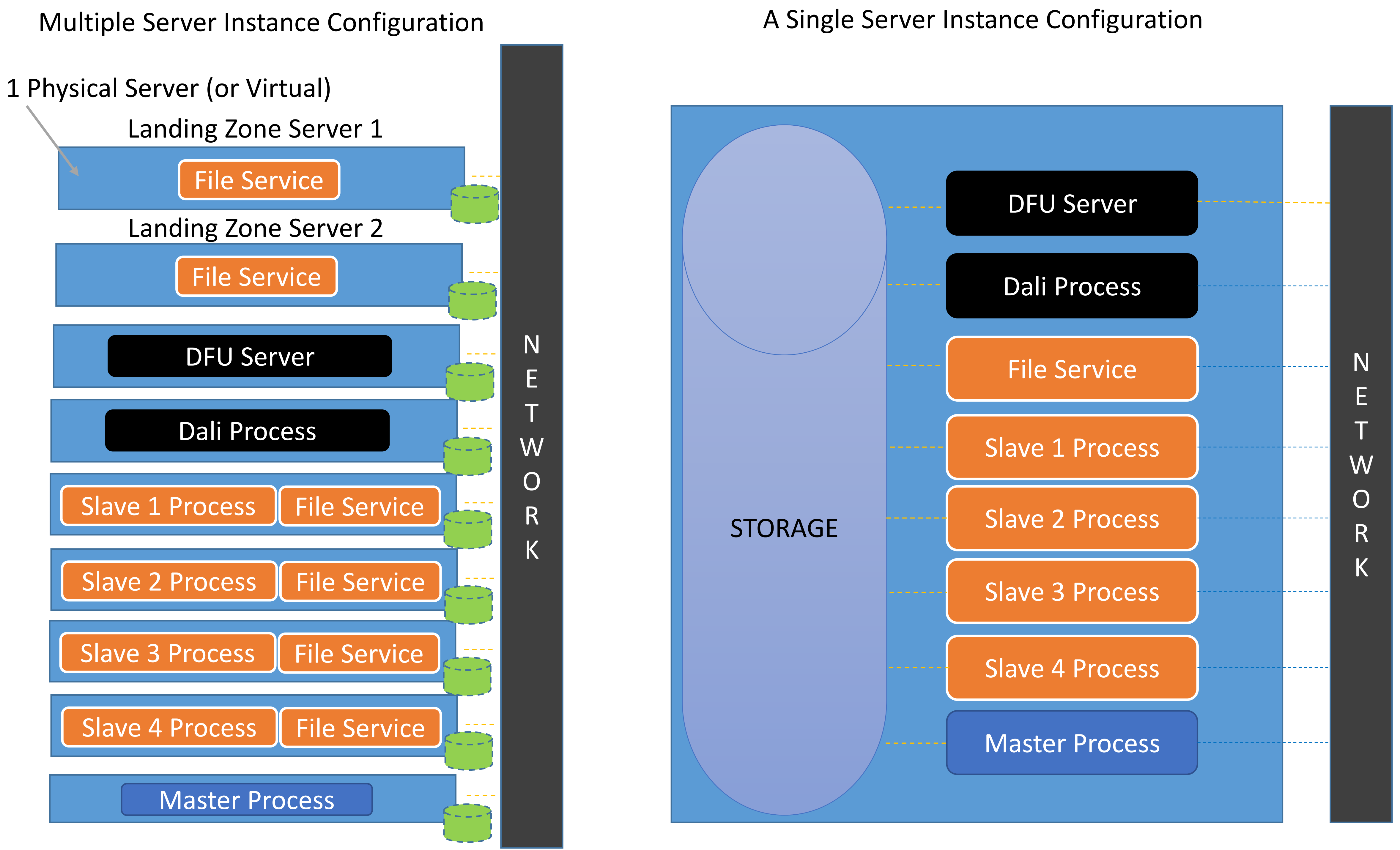


HPCC SYSTEMS®

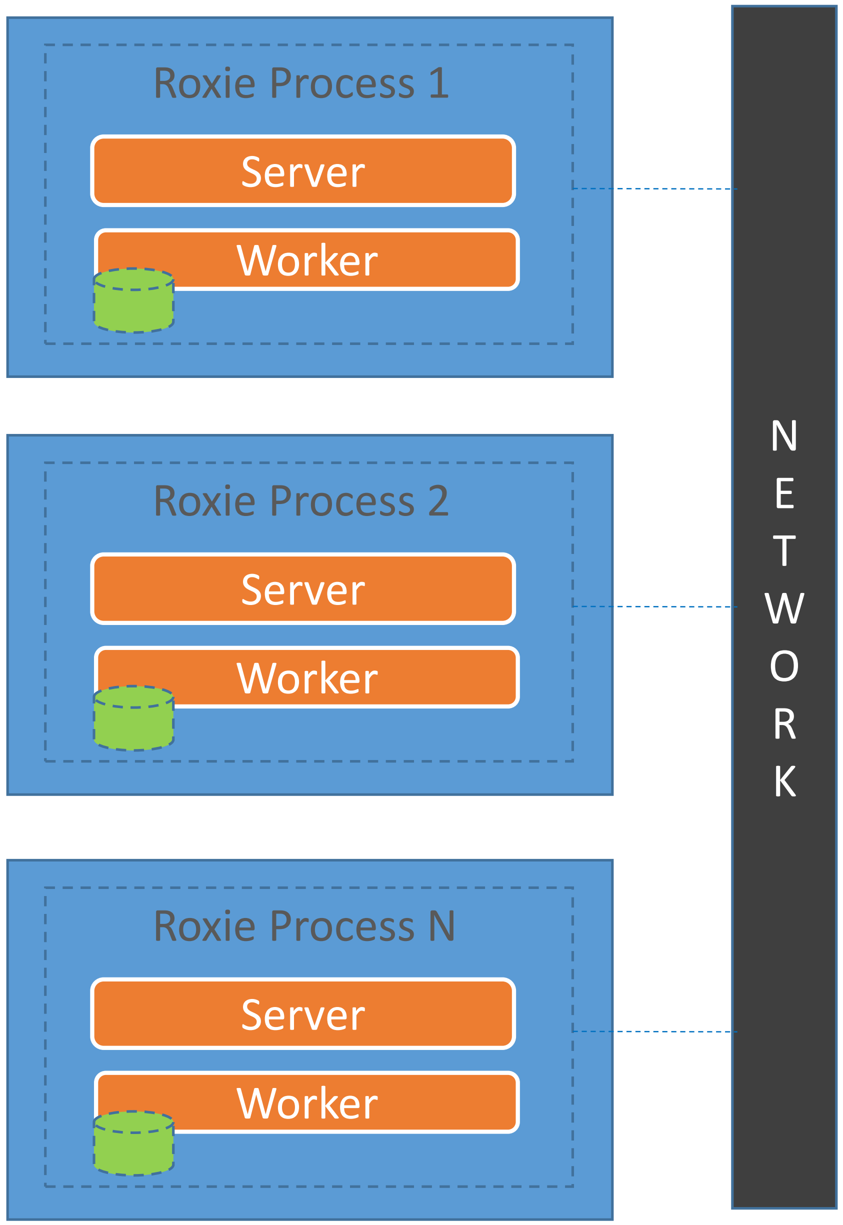


HPCC SYSTEMS®

Thor Cluster


Multiple Server Instance Configuration
1 Physical Server (or Virtual)
Landing Zone Server 1
File Service
Landing Zone Server 2
File Service
DFU Server
Dali Process
Slave 1 Process
File Service
Slave 2 Process
File Service
Slave 3 Process
File Service
Slave 4 Process
File Service
Master Process
NETWORK
A Single Server Instance Configuration
STORAGE
DFU Server
Dali Process
File Service
Slave 1 Process
Slave 2 Process
Slave 3 Process
Slave 4 Process
Master Process
NETWORK

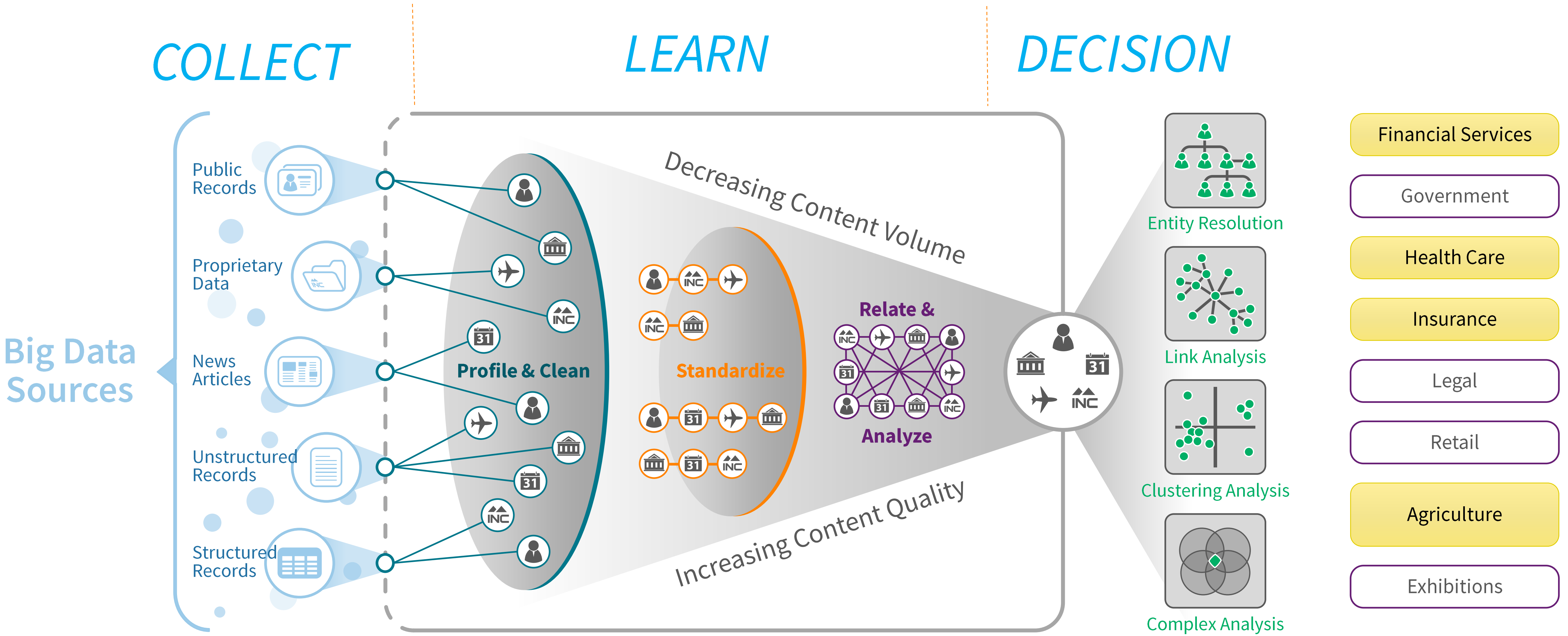
ROXIE Cluster


Roxie Process 1
Server
Worker
Roxie Process 2
Server
Worker
Roxie Process N
Server
Worker
NETWORK

Use Cases

HPCC SYSTEMS®

Insurance Fraud Analytics



Unstructured and Structured Content

- +4 petabytes of content
- 50 billion records
- 10,000 sources
- 7.5 billion unique name and address combinations

High Performance Computing Cluster Platform (HPCC)

Open Source Components

- Cluster Technology
- Data-centric language (ECL)
- Integrated delivery system that offers data plus analytics

Analysis Applications

- Multi-bureau/multi-source models, bureau roll-over support
- Extensive experience leveraging atomic level data, combining disparate data
- ~400 models deployed (custom and flagship)

Key Capabilities

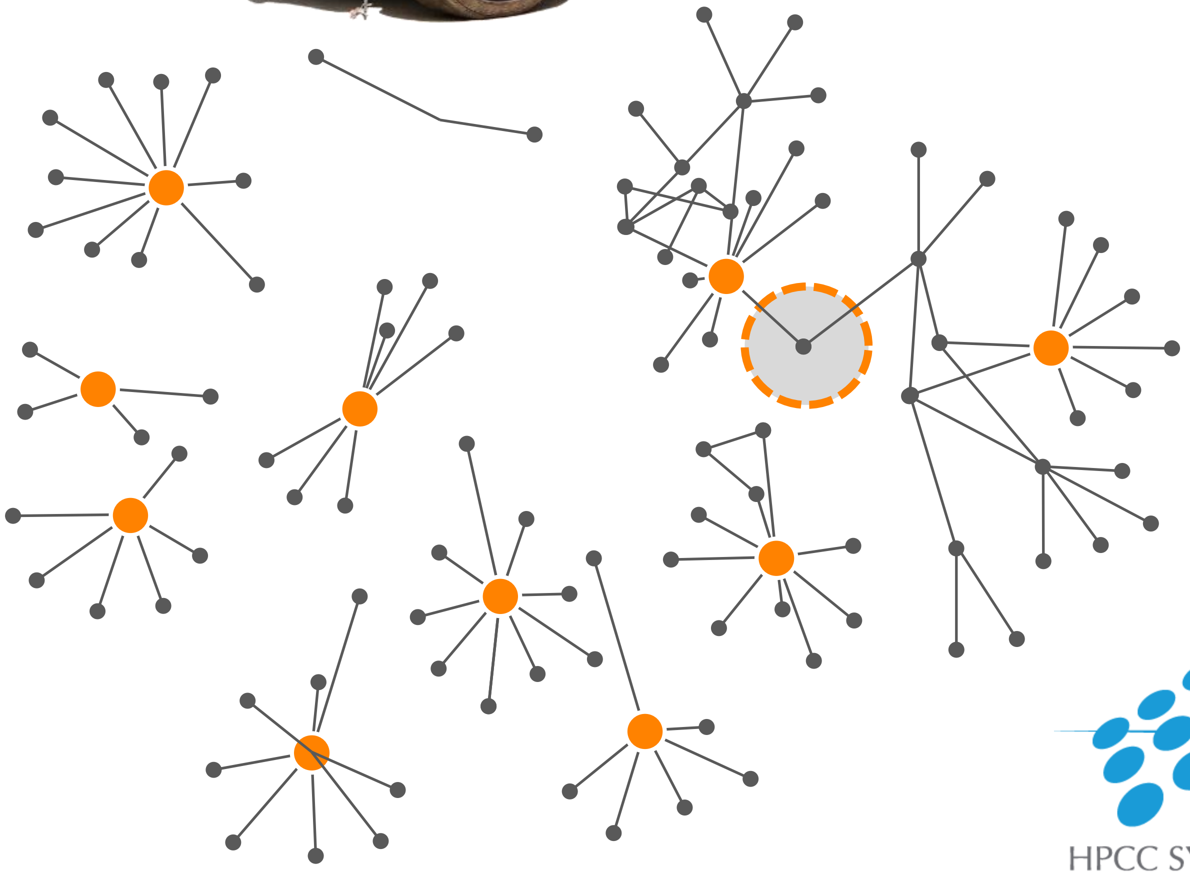
- Data and analytics
- Identity verification and authentication
- Fraud detection and prevention
- Investigation
- Screening
- Receivables management

Use Case: Insurance Collusion

THE CHALLENGE

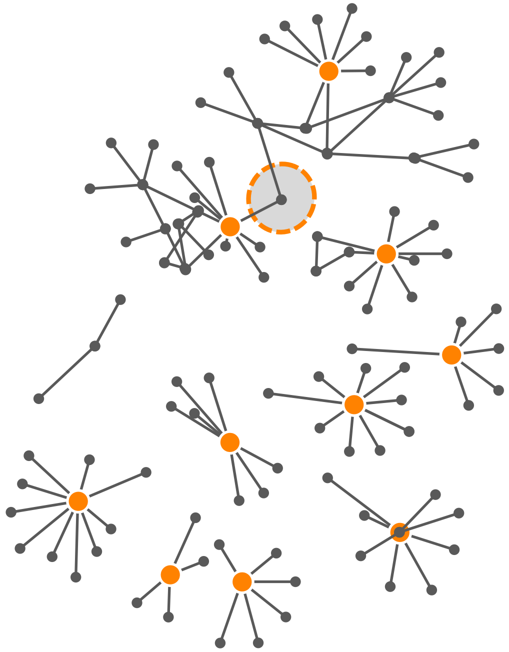
Detecting hidden and comprehensive insurance claim fraud

The Insurance company's data **only found connections between two of the seven claims** — and only identified one other claim as being weakly connected

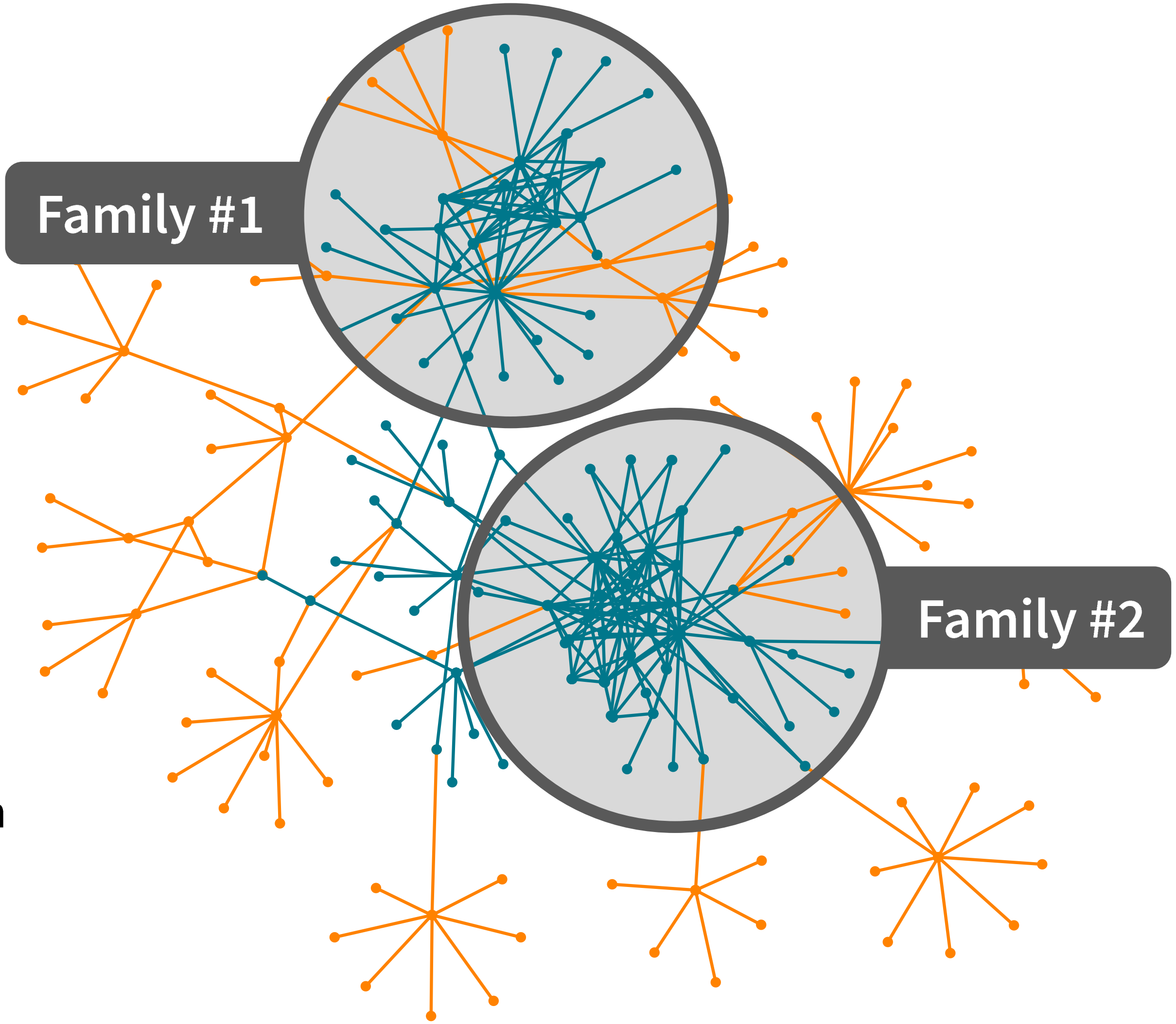


Use Case: **Insurance Collusion**

THE SOLUTION



Customers Claim data is linked with LexisNexis® Risk Solutions data using the **HPCC Systems platform**



RESULTS

- Two family groups interconnected **on all of these seven claims**
- Links were **much stronger** than the carrier data previously supported

Precision Agriculture

Vast amounts of data spread across the Agricultural landscape. Consolidating, organising and enhancing this data to help drive value across the entire industry, from the farm gate all the way to the supermarket shelf.

COLLECT

COLLECT

Farm Management Info Systems
A wide spectrum of tools used by farmers all generating data

Farm Machinery
Every piece of equipment on the farm is now generating data and wants to be precise

Sensors
Ground and animal sensors measure everything from animal fertility to soil moisture

Soil
Global soil type horizons

LEARN

DECISION

Satellites / Drones/ Robots
Ability to identify yield and crop issues from space / drones

Weather Data
Global current and historical weather and soil moisture data at sub-field level

Manufacturers & Distributors
Manage supply chain connectivity between MFRS and their distributors

Agronomist
Providing farm advice, shape files and data to farmers

HPCC SYSTEMS®

Agriculture is at the center of Global Change

Why?

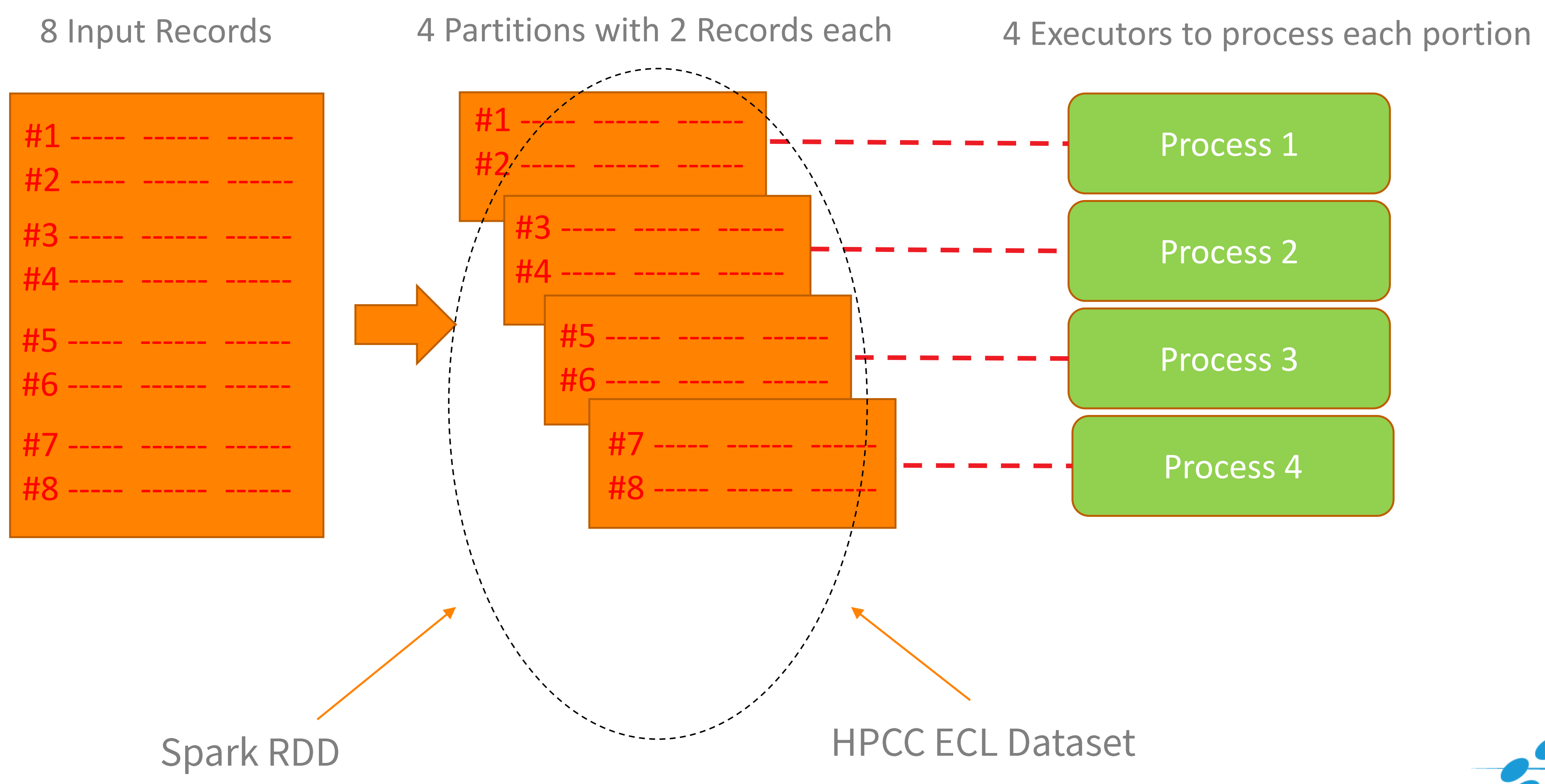
- ✓ Healthier Options
- ✓ Increased Production
- ✓ Decrease Waste

How?

Better Agriculture – Soil Conditioning + Seeding + Fertilization + Irrigation + Pests handling + Harvesting + Weather

Baseline 3 : HPCC Systems and Spark have similar programming models

Data Partitions - Most data operations are inherently parallel



HPCC SYSTEMS®

Description	Spark	ECL
Transform every record to another record	RDD.map()	PROJECT(dataset,…)
Filter	RDD.filter()	dataset(filter)
Transform every record to 1 or more records	RDD.flatmap()	NORMALIZE()
Group records by	RDD.groupBy(), RDD.groupbyKey()	ROLLUP() using GROUP OR DENORMALIZE() using GROUP
Apply Aggregate functions like SUM, COUNT, ETC.	RDD.aggregateByKey(), RDD.reduceByKey()	TABLE() using GROUP
Transforms each element within a partition to multiple or none	RDD.mapPartitions()	NORMALIZE() using LOCAL
Join multiple datasets (or RDDs)	RDD.join()	JOIN()
Distinct records	RDD.distinct()	DEDUP() after SORT()
Repopulate the partitions	RDD.partitionBy()	DISTRIBUTE()
	…	…

HPCC SYSTEMS®

An ECL Transform Operation example using the PROJECT

```
IMPORT STD;

raw_input_record := RECORD
    STRING text;
END;

raw_ds := DATASET([{'JOHN,SMITH,36'},
            {'RAJA,SUNDAR,25'},
            {'MARK,HANFLAND,50'}], raw_input_record);

person_record := RECORD
    STRING50 firstName;
    STRING50 lastName;
    INTEGER  age;
END;

person_ds := PROJECT(
    raw_ds,
    TRANSFORM(
        person_record,
        items := STD.STR.splitWords(LEFT.text, ',');
        SELF.firstName := items[1];
        SELF.lastName := items[2];
        SELF.age := (INTEGER)items[3];
    )
);

OUTPUT(person_ds);
```

Benchmarks

Baseline 4: Test Performed on identical hardware, data and functions

Hardware

- r3.2xLarge Instances
- 8 vCPUs per instance (Each vCPU is 1 hardware thread)
- ~65 GB of memory per instance
- 160 GB SSD storage

Software Setup

- 1 Master Node
- 3 Slave (Executor) Nodes

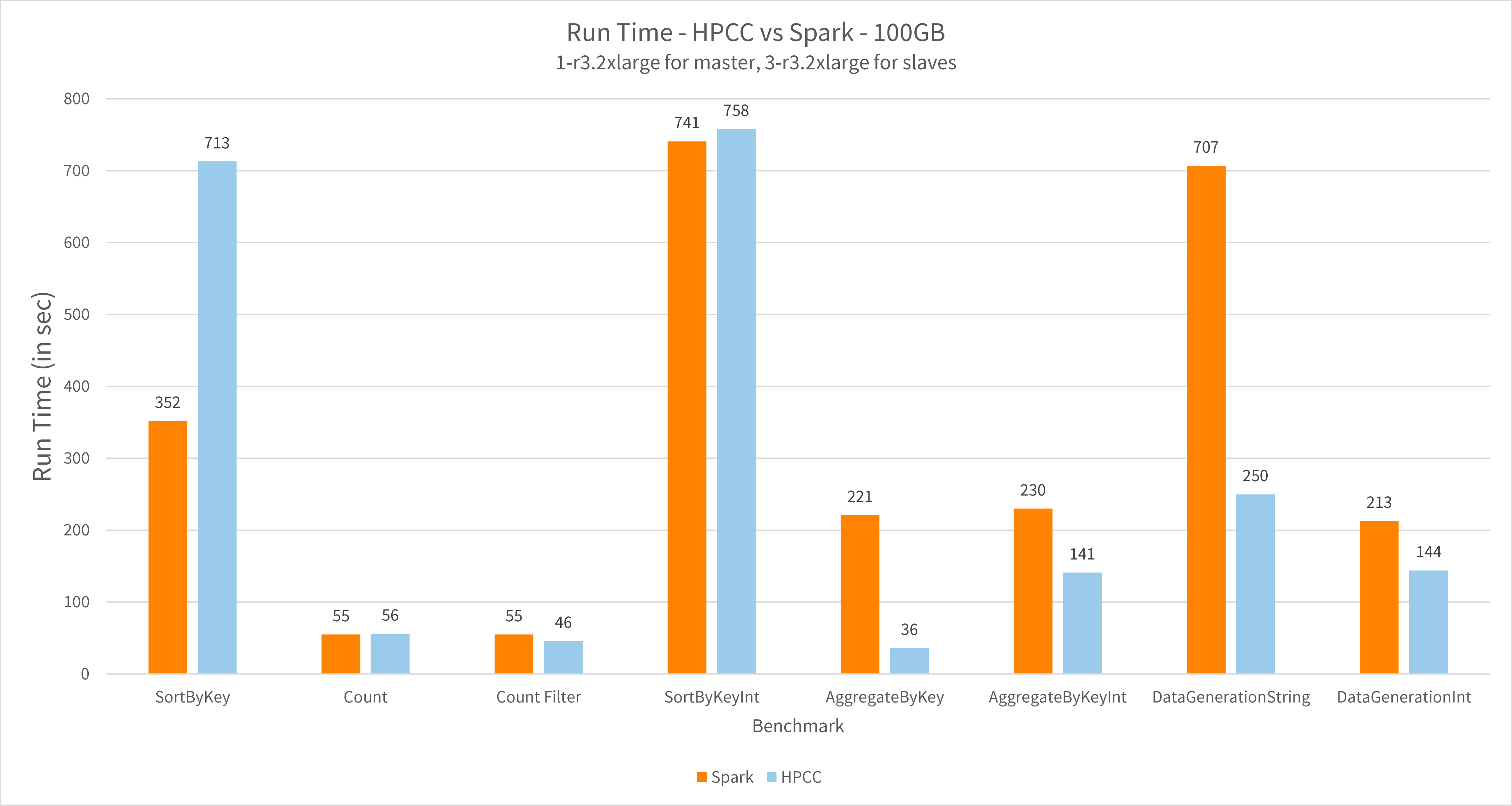
NOTE: Slaves and Executors perform the computation.

Hence, the total available memory is 65 X 3 = ~195 GB

Data

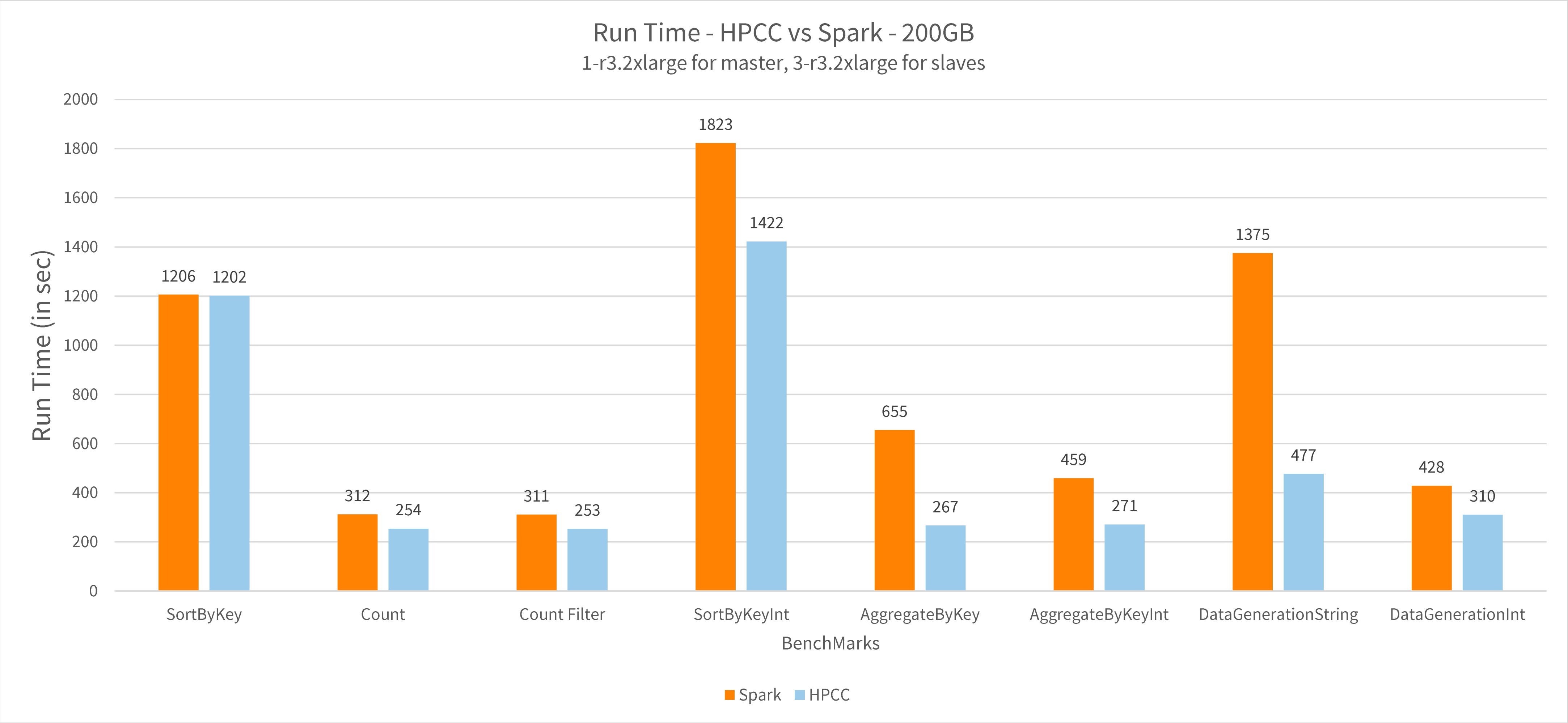
- 100 GB workload
- 200 GB workload

100 GB Test



HPCC SYSTEMS®

200 GB Test



HPCC SYSTEMS®

A shout-out to

- Tim Humphrey – LexisNexis
- Vivek Nair – NC State PhD Student
- James McMullan - LexisNexis

HPCC SYSTEMS®

HPCC SYSTEMS®

Thank You!

End-to-end big data in massively scalable supercomputing platform

Open-source. Easy to use. Proven.

Visit us at hpccsystems.com