

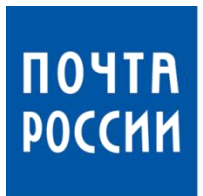
Case Study: Big Data Platform for Russian Post on Hadoop stack



Project by LUXOFT for Russian Post

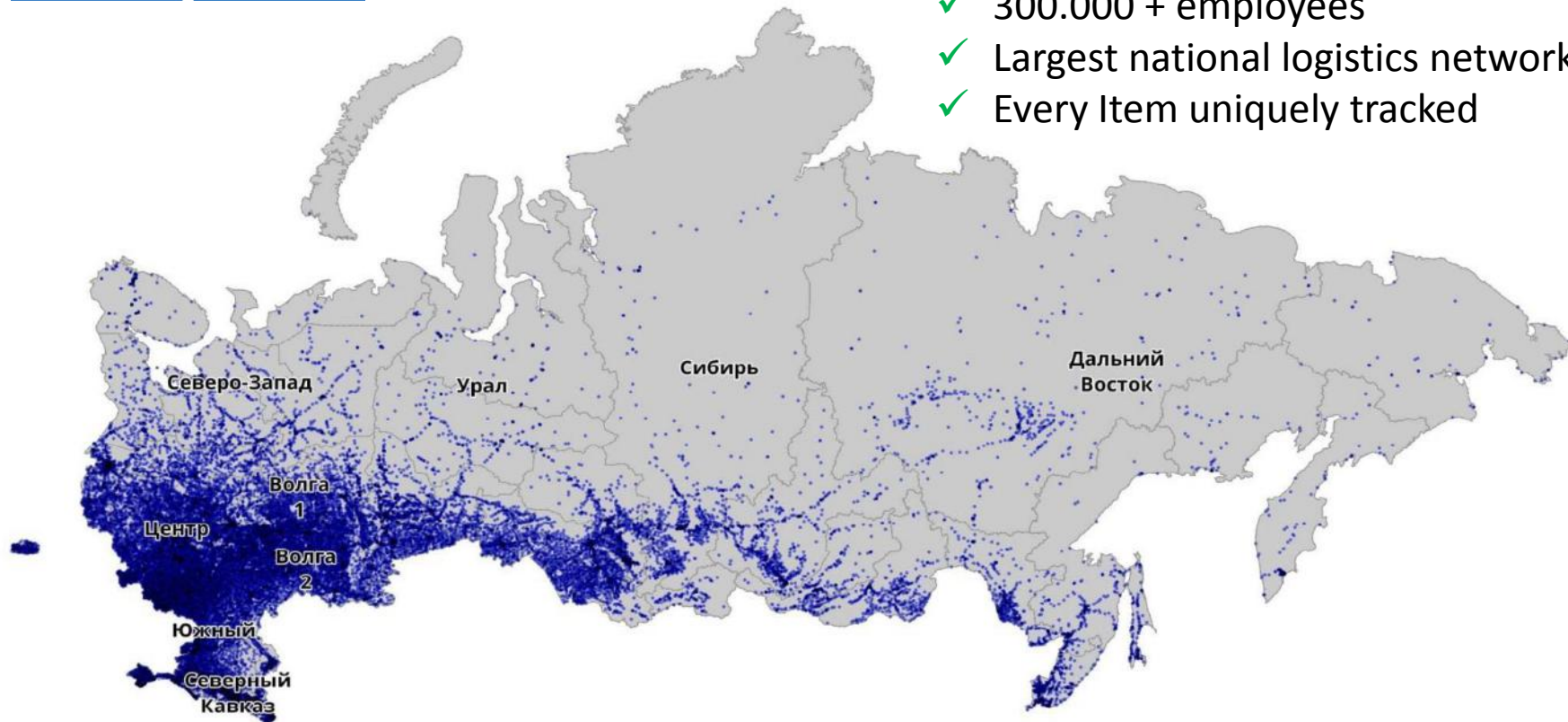


Andrei Bashchenko

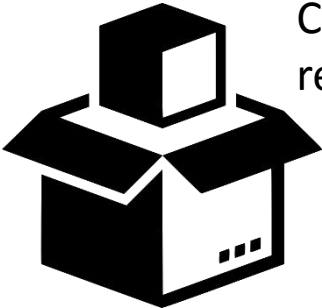


What is Russian Post?

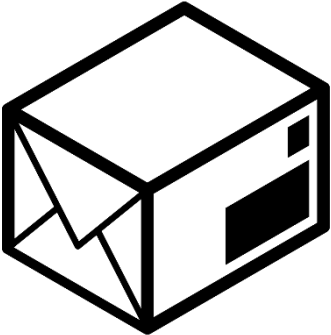
- ✓ 42.000 + retail shops
- ✓ 300.000 + employees
- ✓ Largest national logistics network
- ✓ Every Item uniquely tracked



Data: Registered Post Items



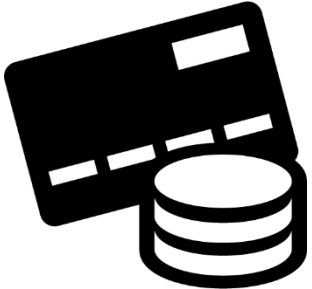
Container-Content relations



Logistics operations



E-com data import



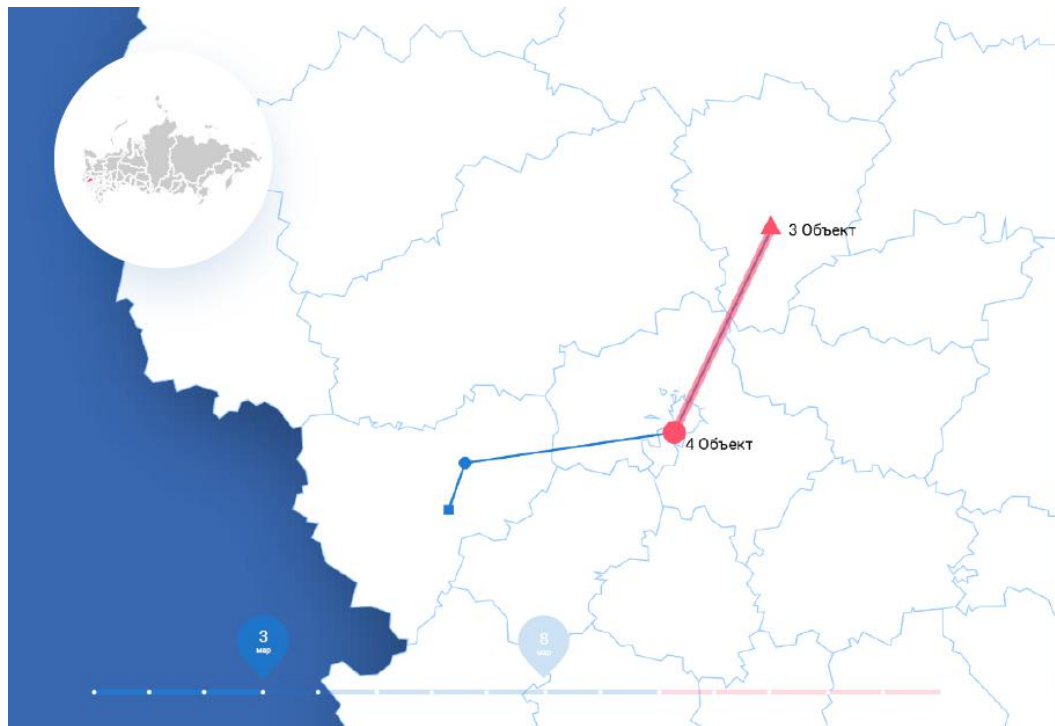
Payments,
Custom Duties



Data: Logistics

- ✓ Routes, Edges
- ✓ Fact vs Control delivery time

- ✓ Transport (fleet utilization, schedule, costs etc.)



- Принято в отделении связи
17 декабря 2015, 14:56 353210, Новотитаровская
- Покинуло сортировочный центр
18 декабря 2015, 12:27 350964, Краснодар
- Сортировка
20 декабря 2015, 23:46 140983, Львовский
- Покинуло сортировочный центр
23 декабря 2015, 05:11 390910, Рязань
+1 д
- Прибыло в место вручения
23 декабря 2015, 13:43 390048, Рязань
+3 д
- Возврат
26 декабря 2015, 14:16 390048, Рязань
+4 д

Logistics operations: no data locality

- ❌ No geo locality (operation in any region could affect items going from and to any other regions)
- ❌ No time locality (operations can arrive months late)



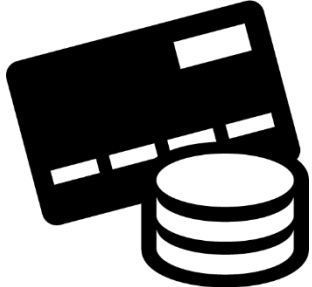
Data: Retail operations



Acceptance,
Delivery



Retail
purchases



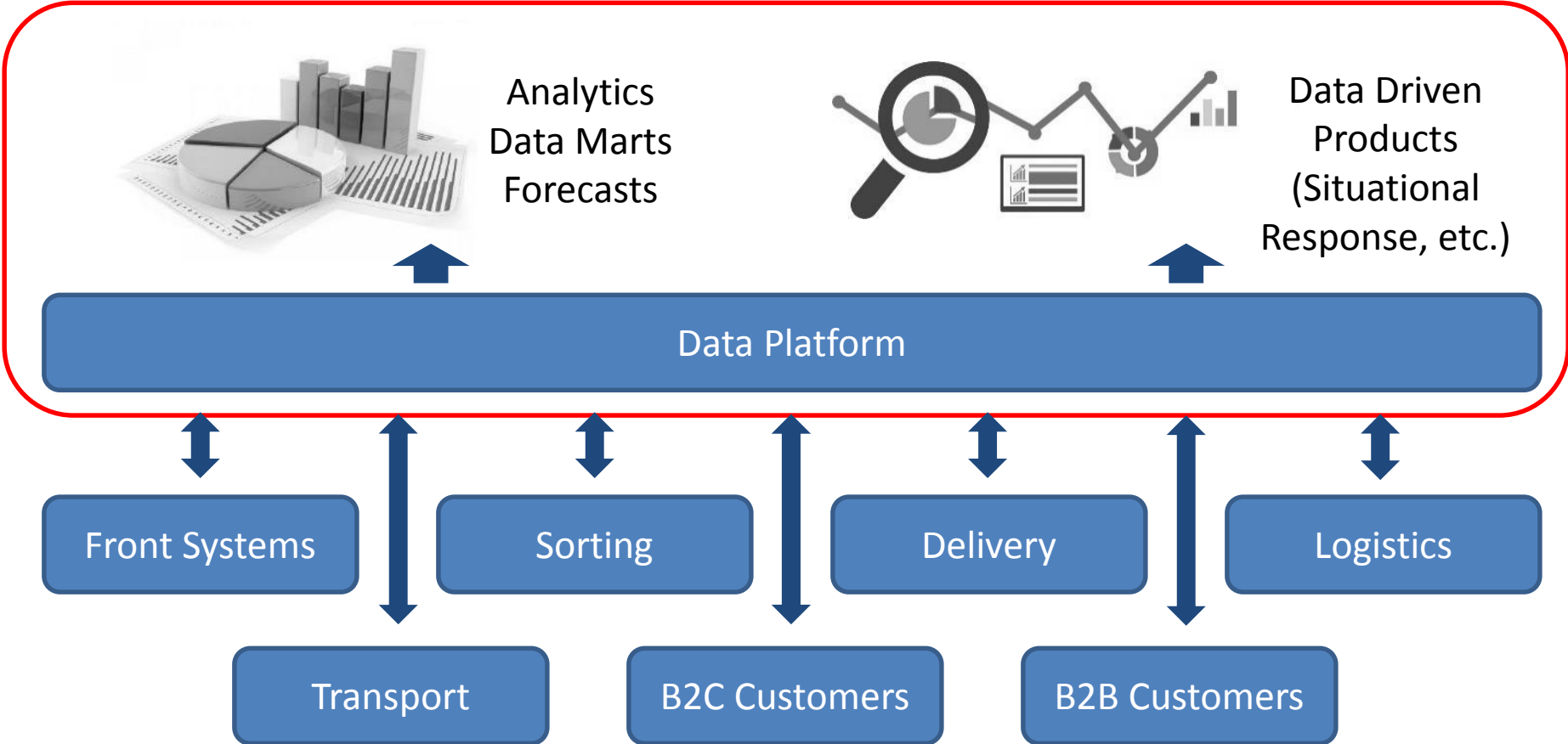
Payments



Transfers



IT Landscape



Meet the Project

Before:

- ✓ Previous solution was Exadata + Oracle BI and covered only batch analytics

Project started in 2015. Goals :

- ✓ Provide batch and real-time analytics and data for all users and systems
- ✓ Do it with open source software (**with one exception**)
- ✓ Ensure horizontal scalability on commodity hardware

Now (2018):

- ✓ Team is 34 professionals: Architects, Software Engineers, Team Leads, Analysts, QA Engineers, DevOps.
- ✓ Processing 100 bln of events per day
- ✓ 1+ Pb of data
- ✓ 7200 VCPU, 25 Tb RAM, 2,5 Pb HDD

Business Tasks example for Batch Processing

«What gets measured gets managed.» (C) Peter Drucker

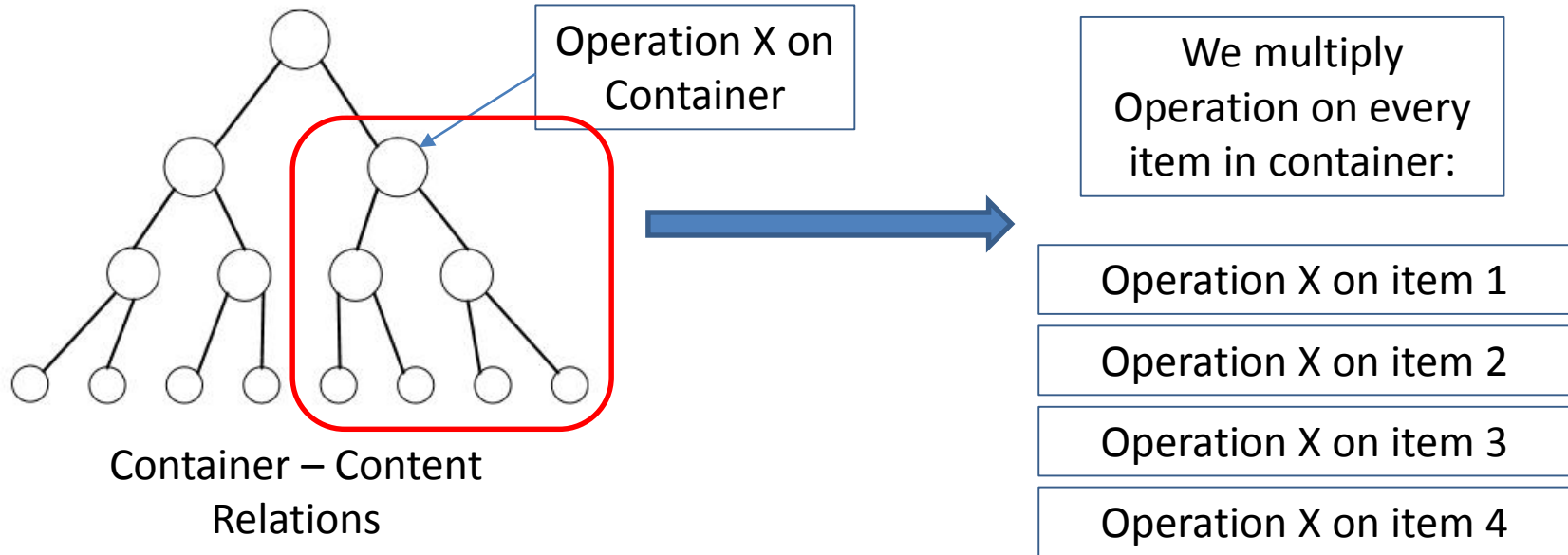
Provide following analytics for Logistics KPI's (with drill-down to a single post item):

1. Post Items balances in every Object
2. Fact vs control delivery time
3. Post Items processing speed in every Object

And many more.

How do we solve this tasks?

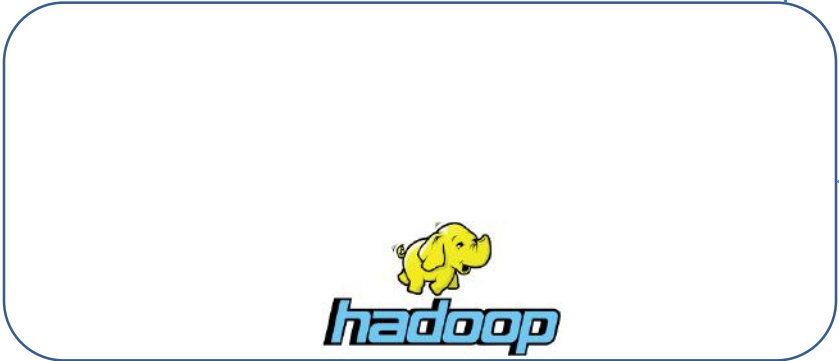
- ✓ We need a collection of all operations on all items
- ✓ Collection of 100 bln events to be recalculated daily
- ✓ Appr 100 attributes per operation
- ✓ We need to 'unfold' operations on content items



Interface



Rest
In



Processing and Storage



Interface

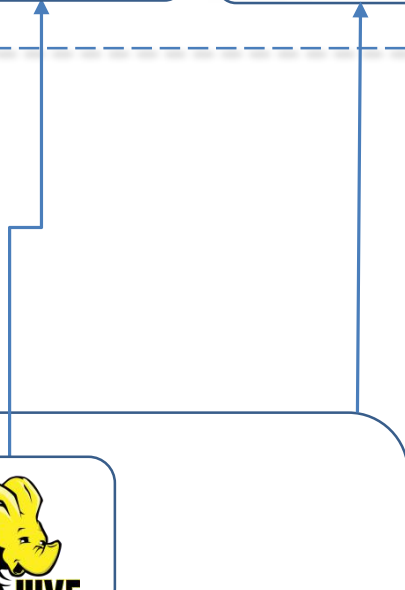


Map-Reduce ~~X~~

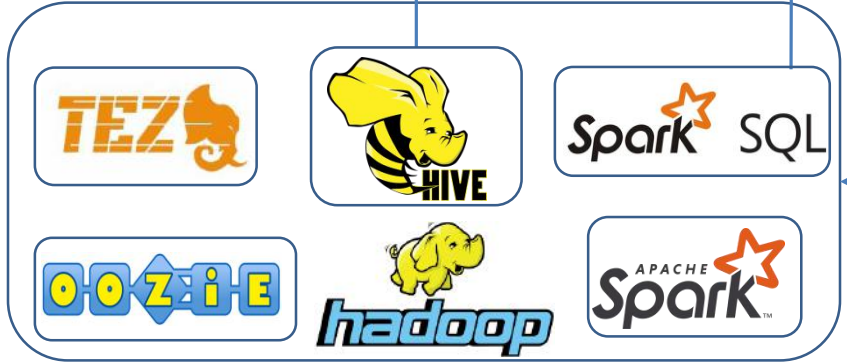
hadoop



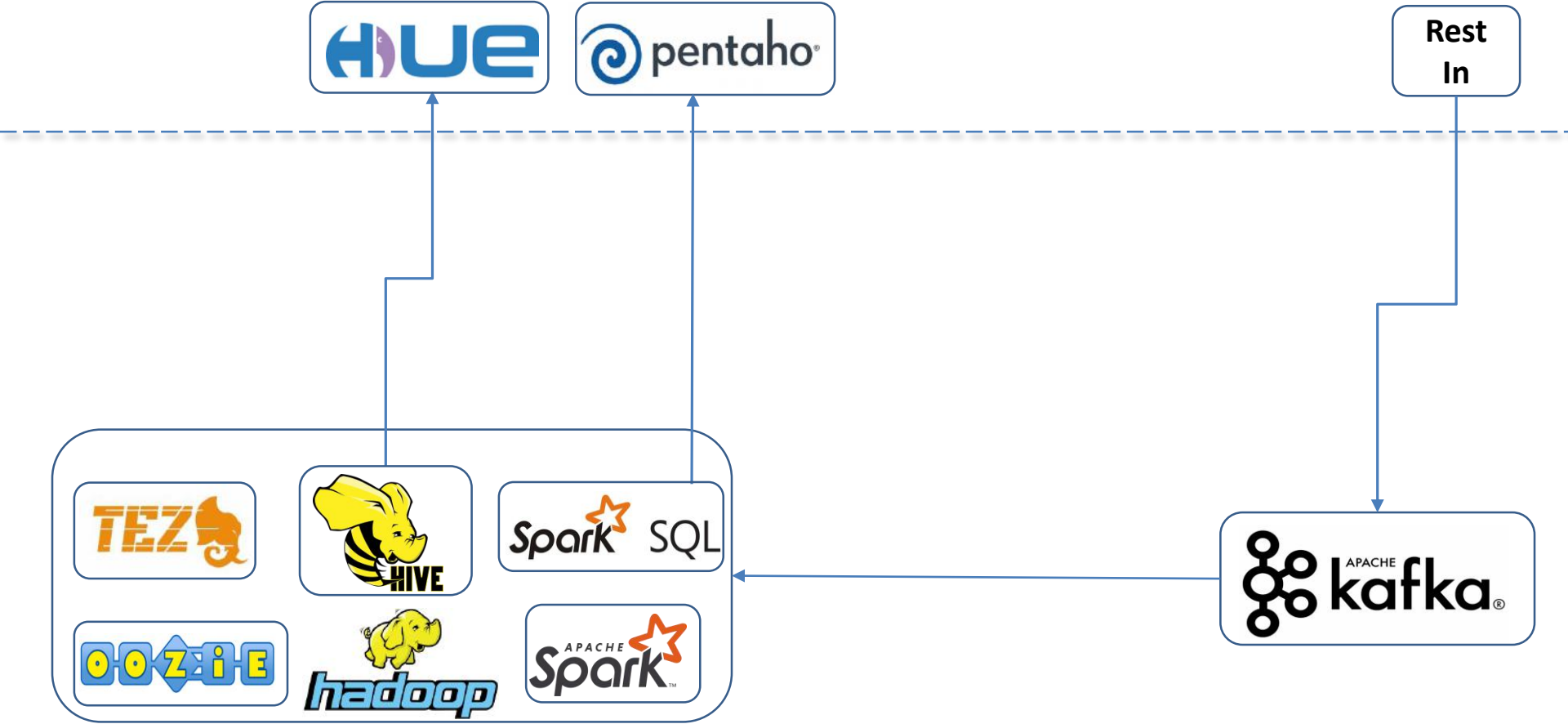
Processing and Storage



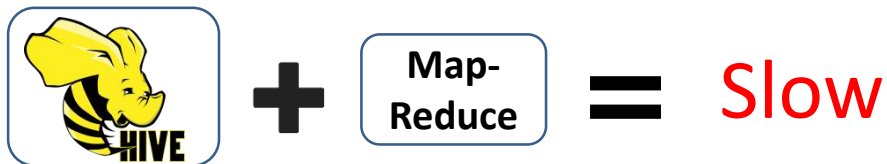
Interface



Processing and Storage



Batch Processing Highlights



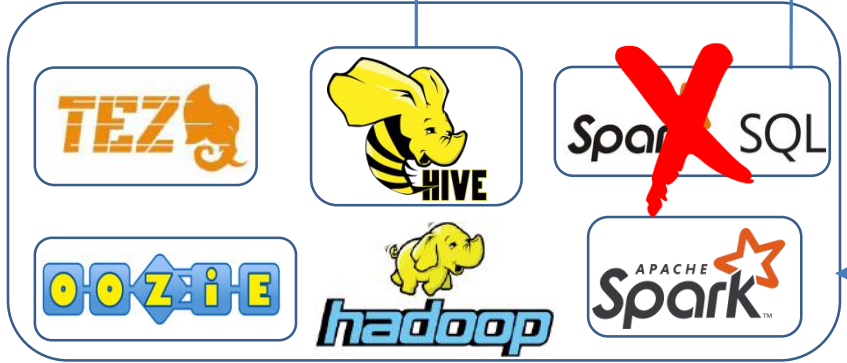
Non-relational
transformation logic ?



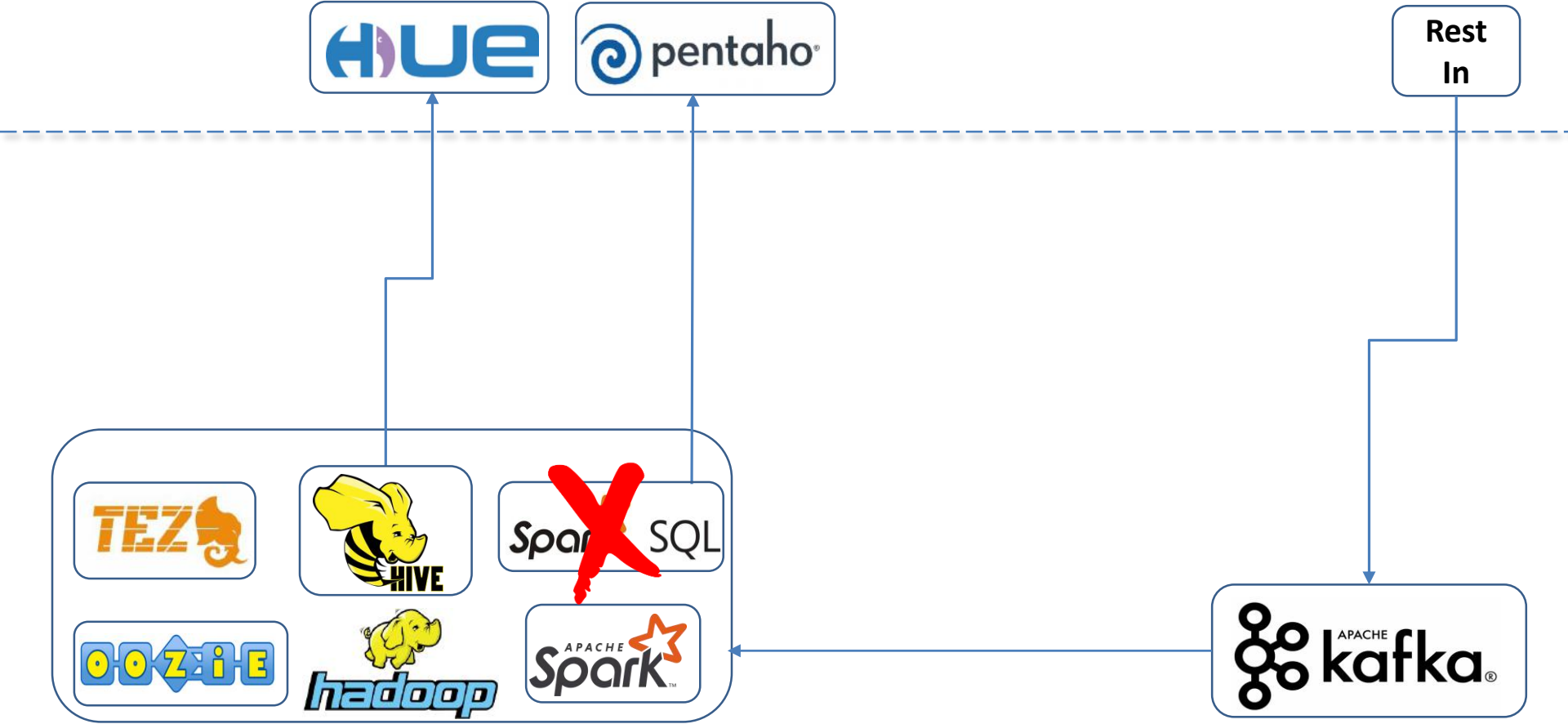
Use



Interface



Processing and Storage



Column storage choice

Use Case: We need highly available column storage for calculated data marts to serve BI requests. It should provide:

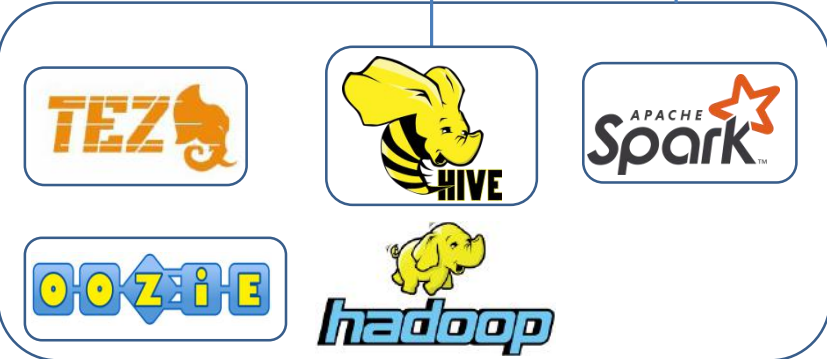
- ✓ Fast data marts load from HDFS
- ✓ Effectively serve big number of simple BI queries– from 300.000 users
- ✓ Effectively serve complex OLAP-like queries



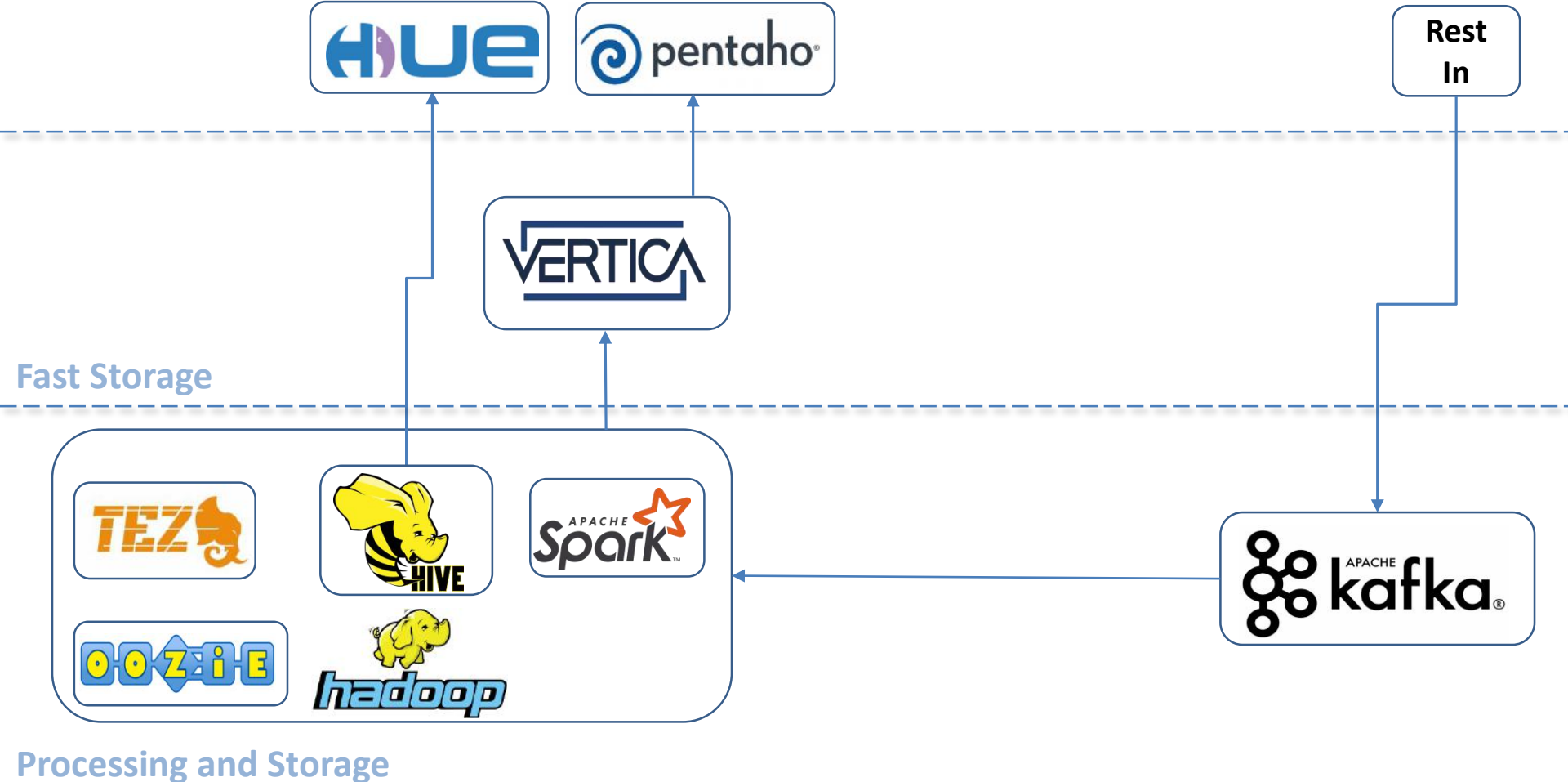
Interface



Fast Storage



Processing and Storage



New tasks: access by key

- ✓ From 100 bln collection of operations we need to get up to 100 records by key
- ✓ Solution should provide quick (<1 sec) response and serve min 100.000 requests per day




Отслеживание

Введите трек-номер или несколько номеров через пробел

[Как узнать трек-номер](#) · [Розыск отправлений](#)

В пути

Письмо из Перу 



Почта России
Прошло регистрацию
30 сентября 2018, 09:56 102976, Шарлово

Прибыло в сортировочный центр
30 сентября 2018, 01:51 102975, Шарлово

Прибыло на территорию России
30 сентября 2018, 01:51 102976, Шарлово

Почта Перу
Ожидает отправки из Перу
22 сентября 2018, 10:22 PELIMA, Перу

Прибыло на границу Перу
22 сентября 2018, 09:34 PELIMA, Перу

Письмо - 45 г
Кому: 
Куда: Москва, 

Interface



Spark SQL ~~thrift~~

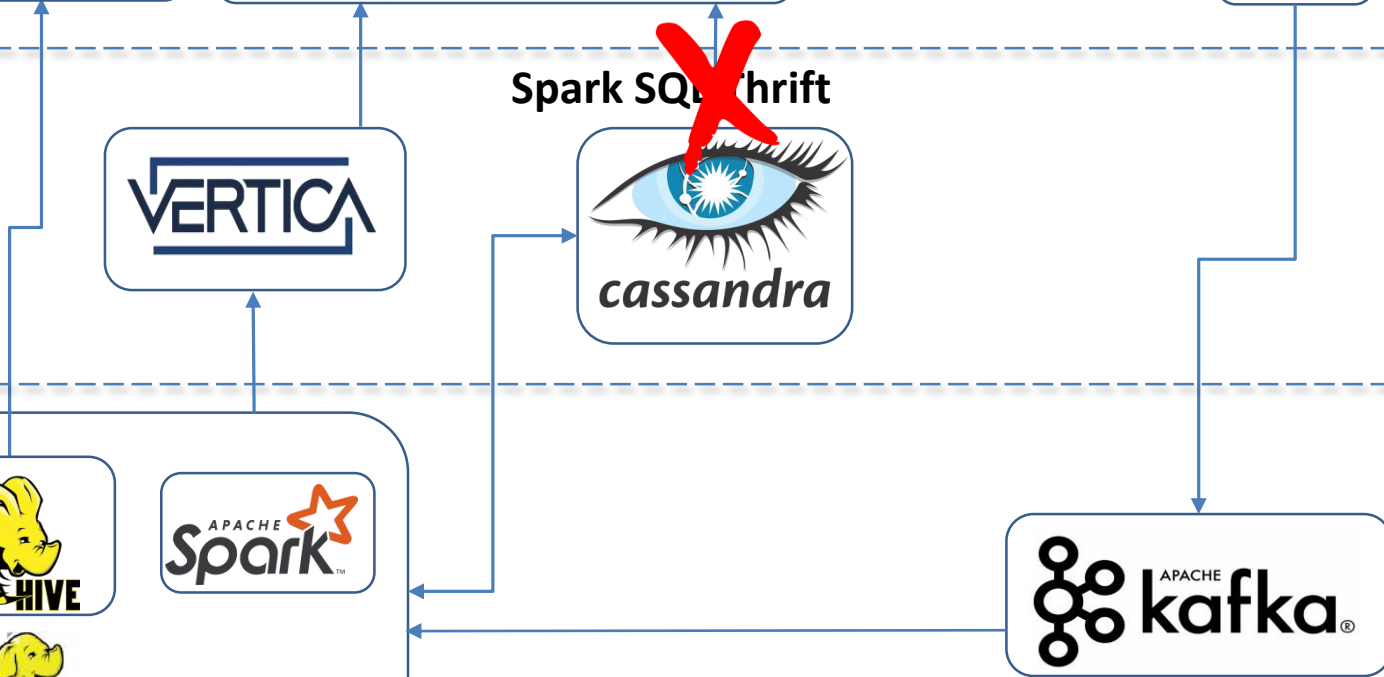


Fast Storage

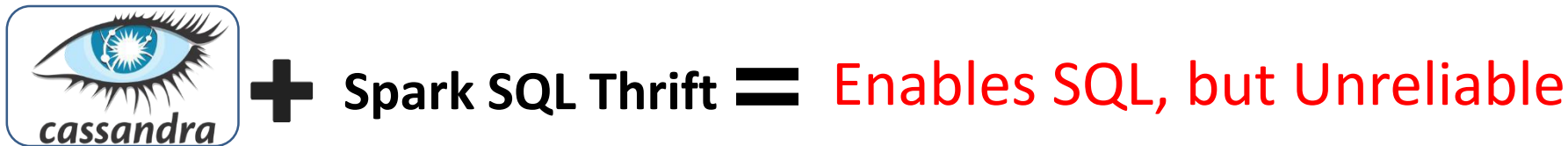
A rounded rectangular container holding five logos: TEZ (orange), HIVE (yellow bee), SPARK (orange star), OOZIE (blue and yellow), and HADOOP (blue elephant).



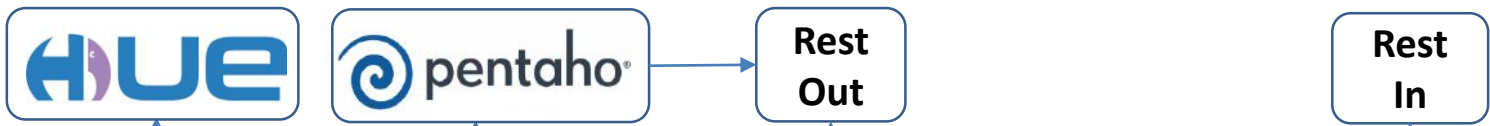
Processing and Storage



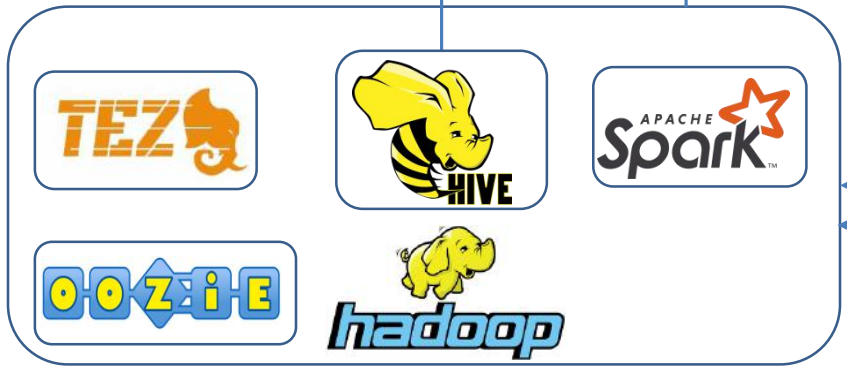
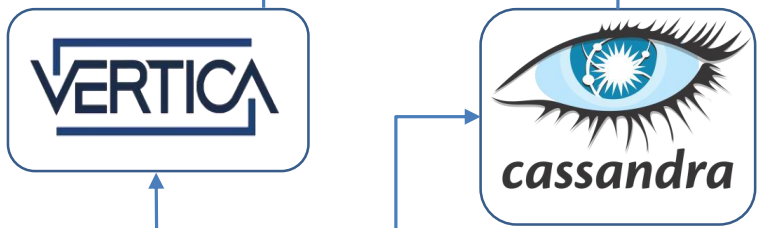
Cassandra Highlights



Interface



Fast Storage

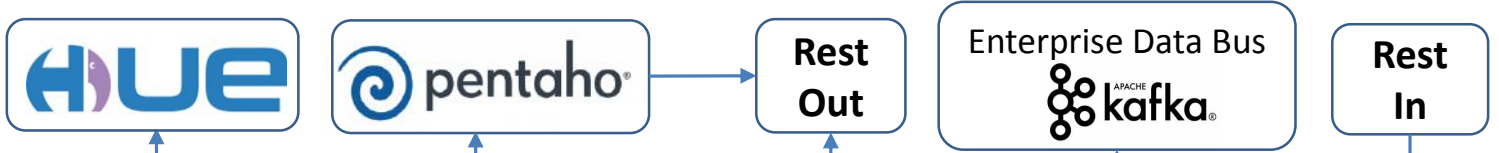


Processing and Storage

New tasks: streaming processing

1. Provide all systems with real-time data they need:
 - ✓ Process real-time data streams
 - ✓ Integrate and enrich data from all systems
 - ✓ Provide actual state data (golden records) on entities by request
2. Provide business users with real-time data in reports (λ -architecture)
3. Provide capability for real-time situational response

Interface



Fast Storage



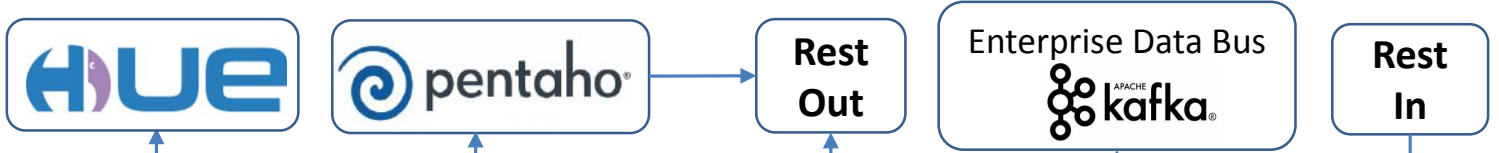
Processing and Storage



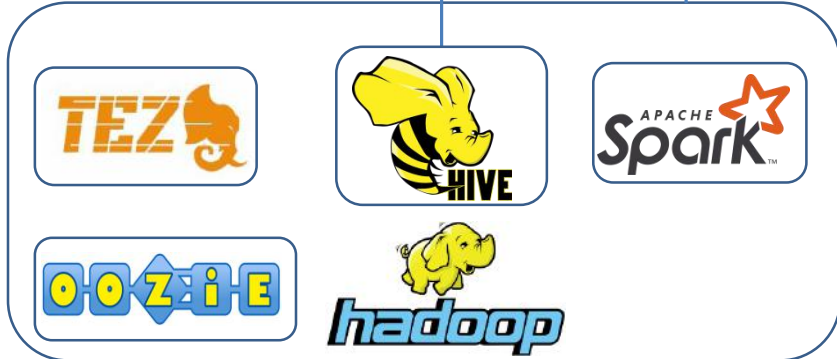
Streaming Highlights

- ✓ When spark streaming is running on the same HDFS cluster used for batch processing:
 - ✓ We need maximum resources available for batch processing
 - ✓ If streaming job fails during heavy batch processing – it waits for resources
 - ✓ Solution – run spark streaming cluster separately (we chose separate HDFS cluster)
 - ✓ Alternative – reserve resources using Yarn
- ✓ Streaming and batch data should match
 - ✓ Use Flume to write streaming data to HDFS

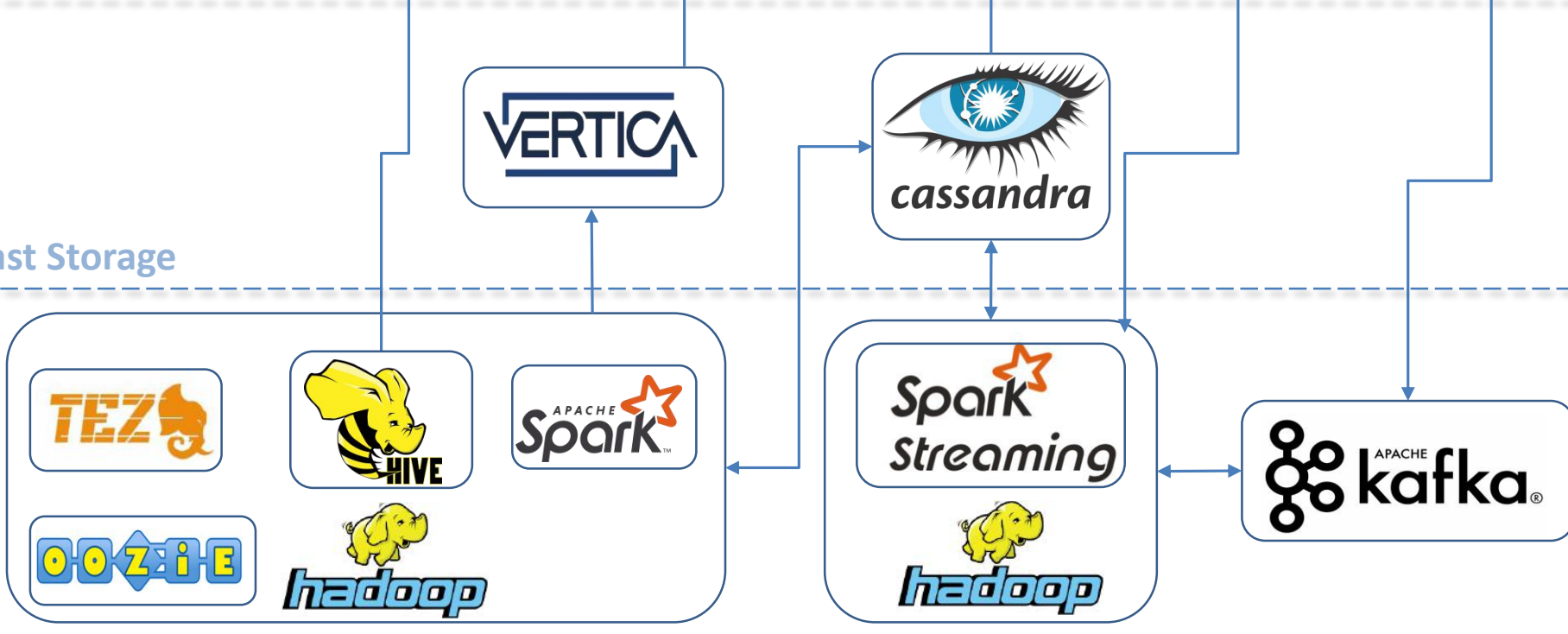
Interface



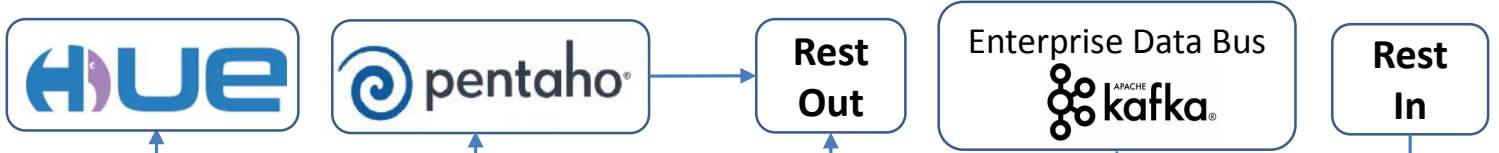
Fast Storage



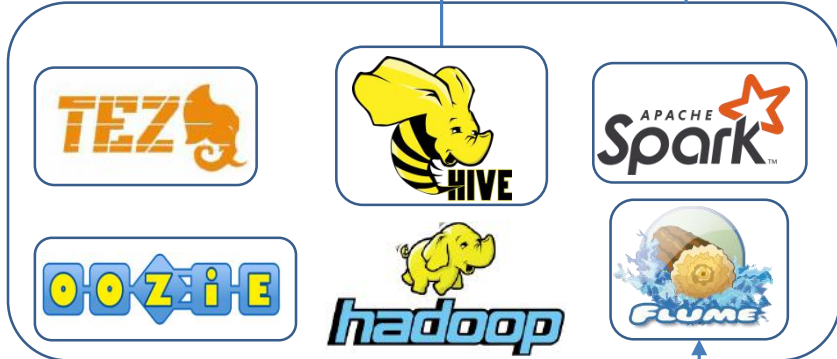
Processing and Storage



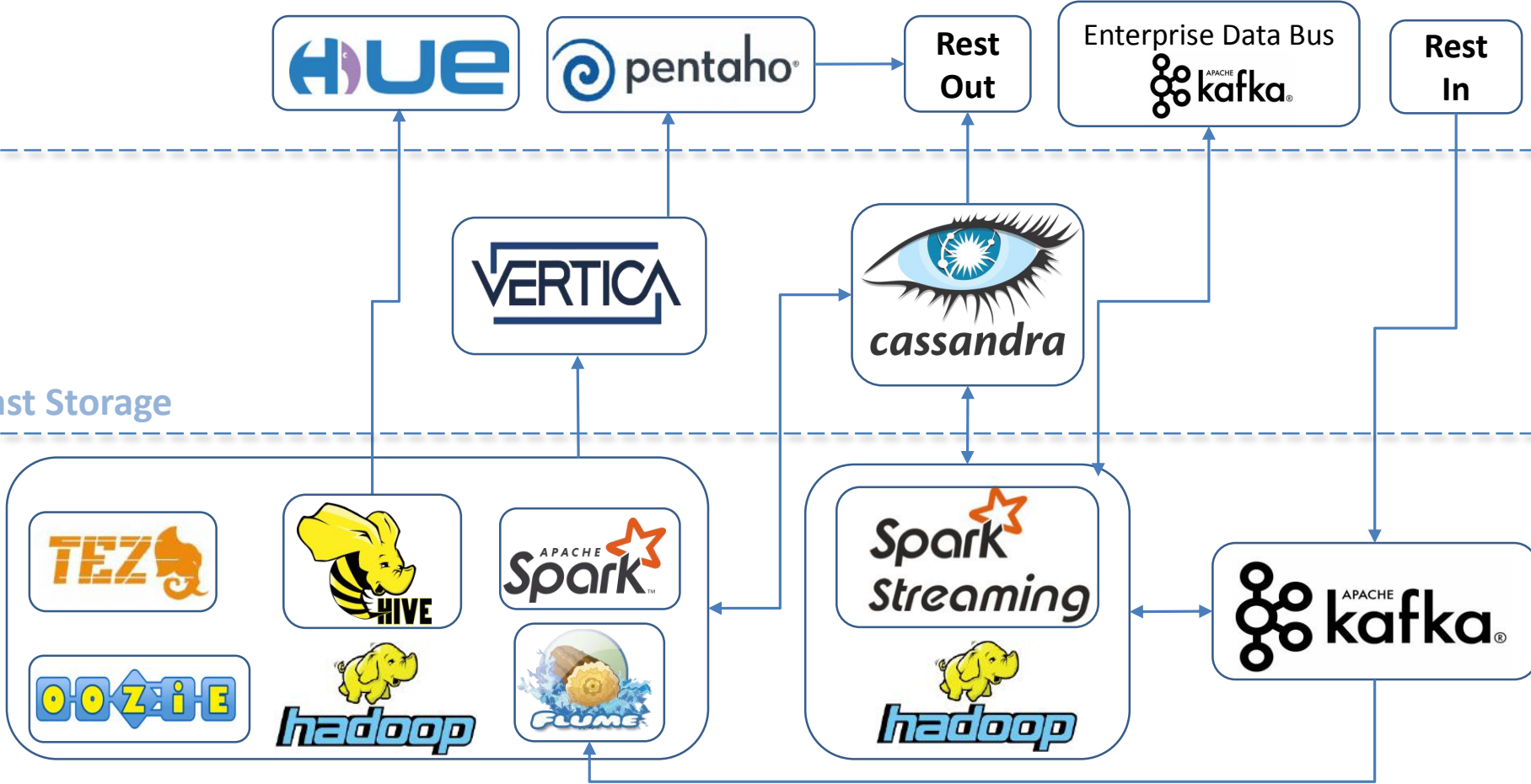
Interface



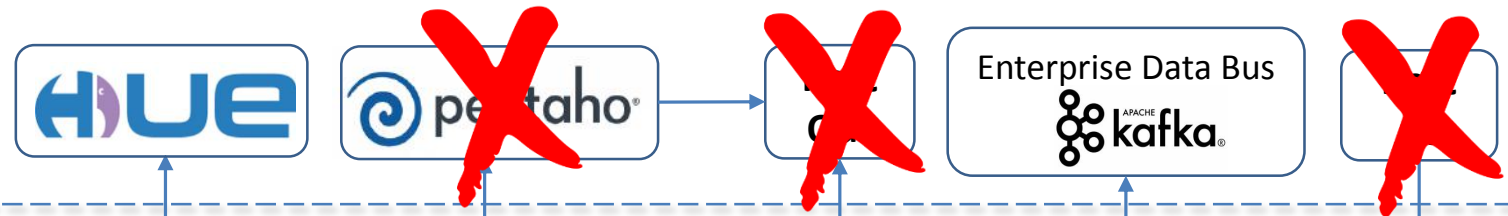
Fast Storage



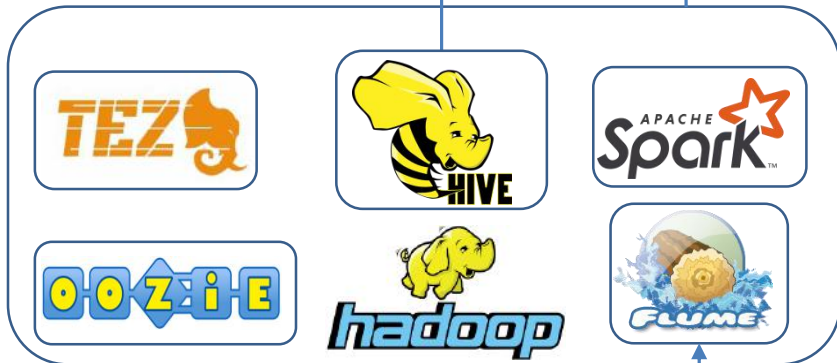
Processing and Storage



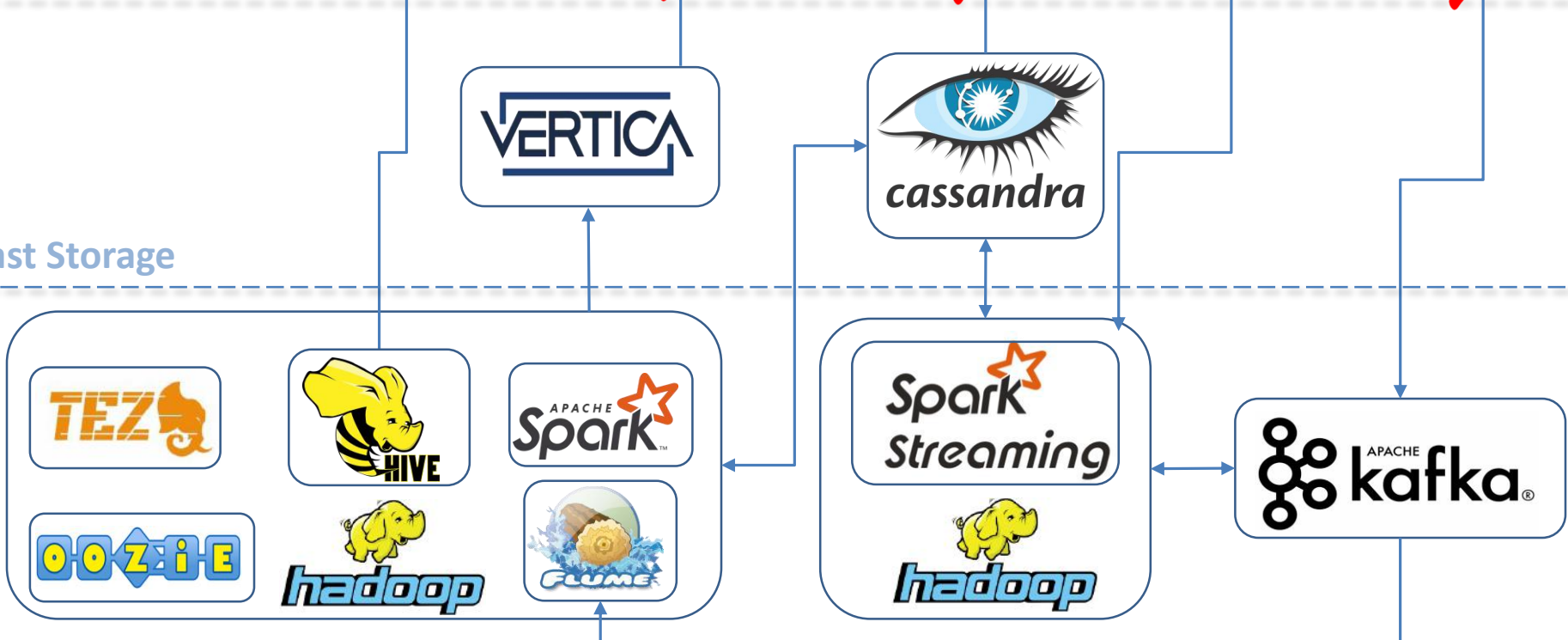
Interface



Fast Storage



Processing and Storage



Interfaces high availability and load balancing

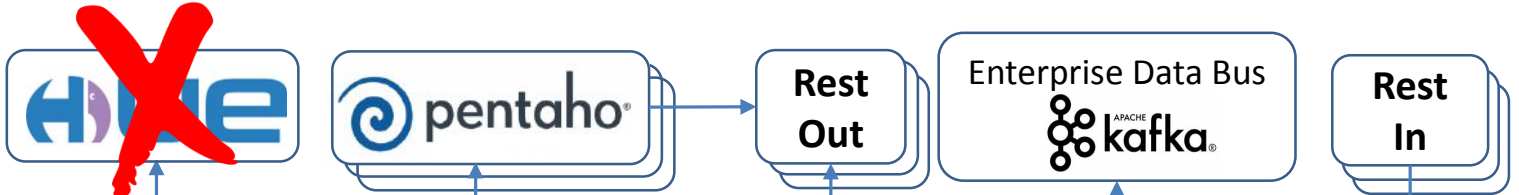
Problem: Need high availability for critical interfaces:

- ✓ BI Pentaho
- ✓ Rest Services

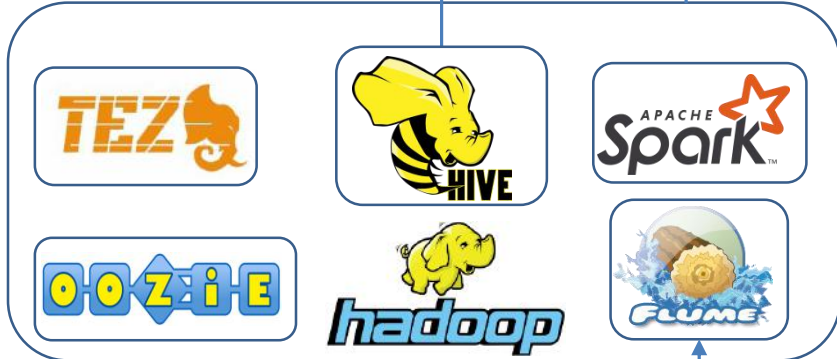
Solution: Docker Containers + load balancer



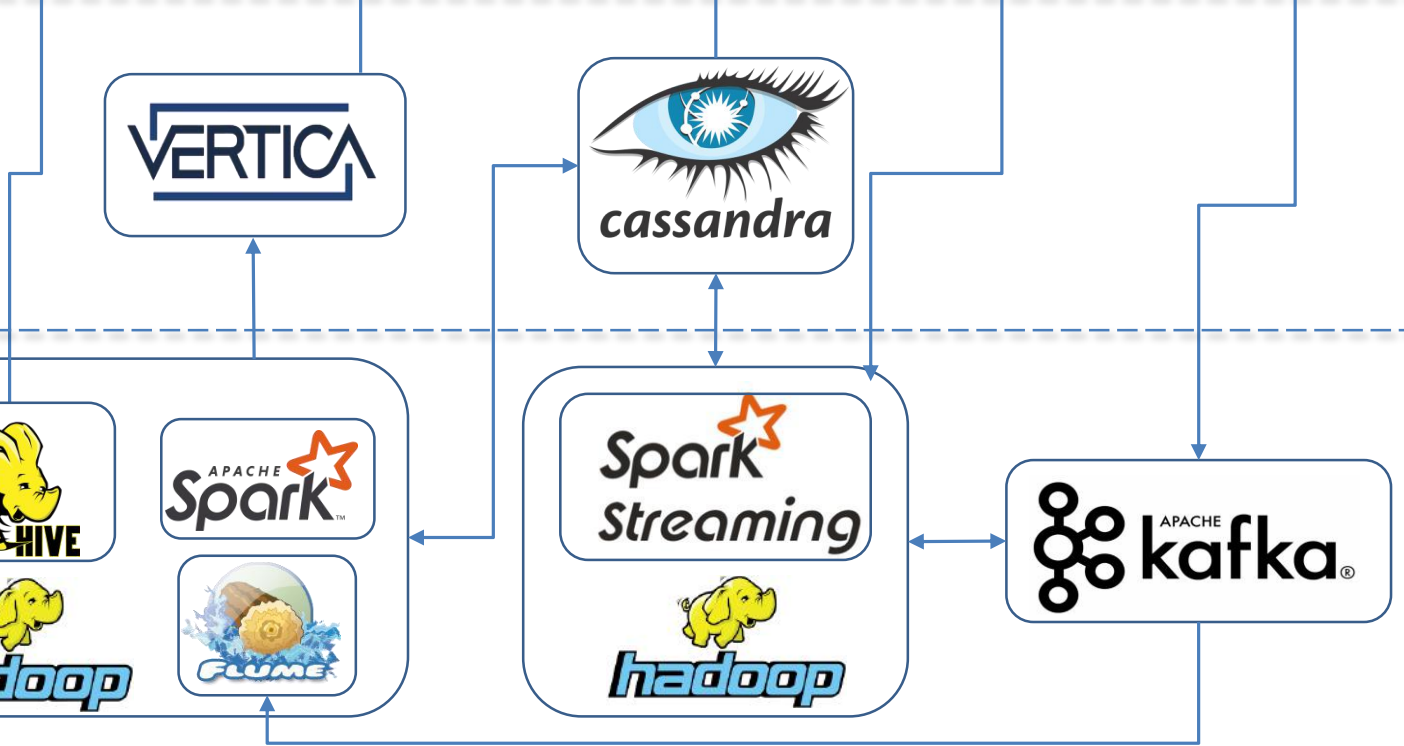
Interface



Fast Storage



Processing and Storage



New tasks: Data access – free analytics

Task: business demand SQL access to data marts.

Limitations:

- ✓ Queries submitted via Hue to Hive could affect performance
- ✓ Users SQL access to Vertica could affect performance
- ✓ Vertica is proprietary solution, price per volume

Solutions:

- ✓ ClickHouse – fast and open source
- ✓ Separate HDFS and Hive cluster, no jobs running, user queries only

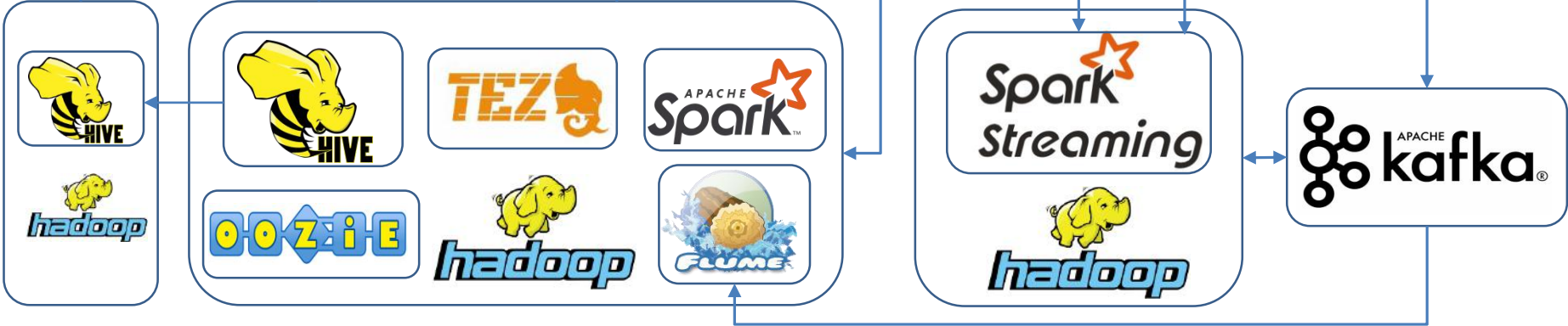
Interface



Fast Storage



Processing and Storage



Data-driven product on top of platform

– Situational Response case



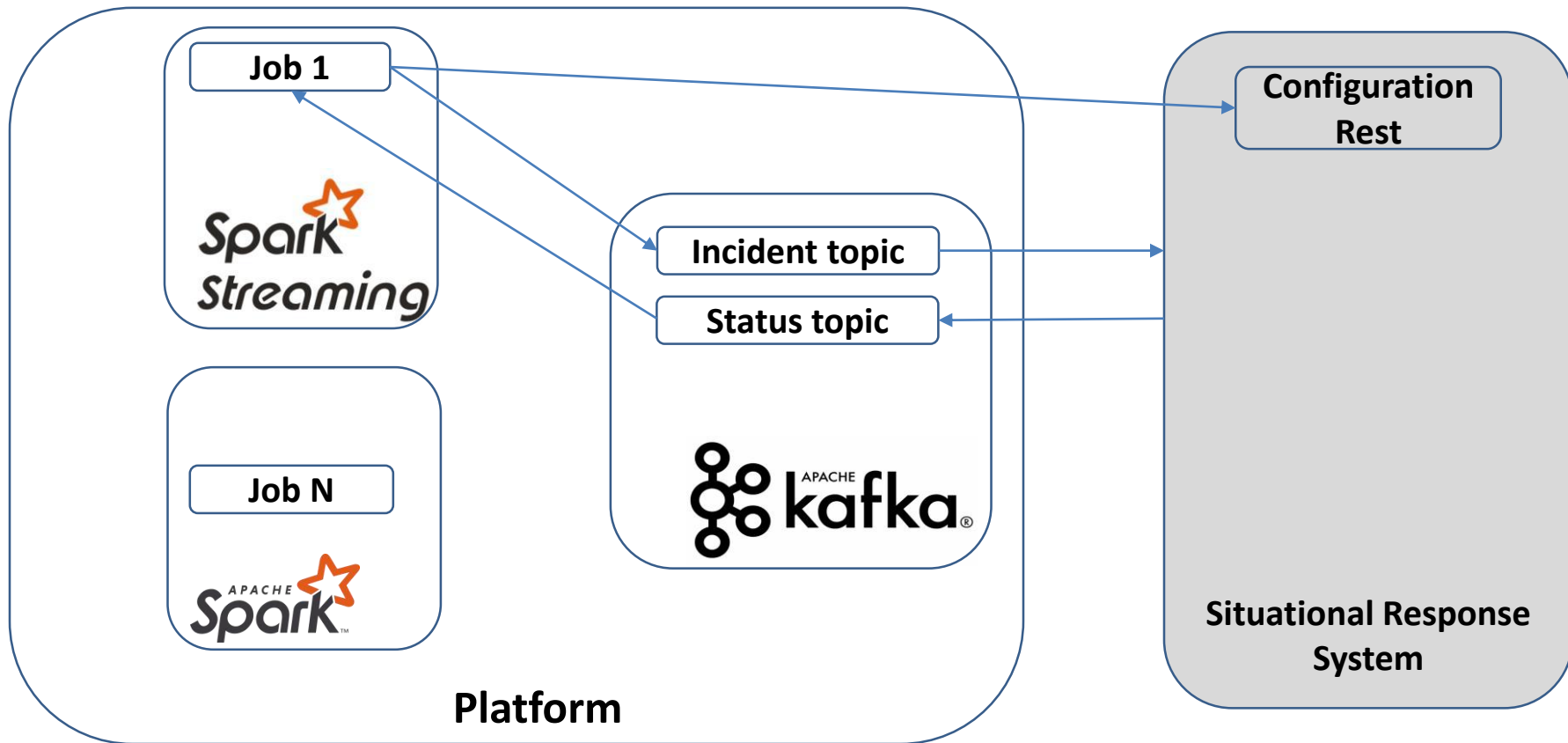
Identification of incidents, that require real-time resolution:

- ✓ Incident tickets creation based on streaming and batch data analysis
- ✓ Data analysis is performed by triggers – streaming or batch jobs, checking data against specific patterns
- ✓ Triggers could encapsulate complex logic, heuristic algorithms
- ✓ Automatic response actions could be setup to resolve incidents



Incident example – item cycling

Data-driven product integration with platform - example



Previous presentations:

✓ By Alexey Vovchenko on Solution Architecture (Highload++ 2017):

https://www.youtube.com/watch?v=6JwMho6UfK4&index=115&list=PLH-XmS0ISi_yn4pCZVOHqWaqQ9trMpyLI

✓ By Andrei Bashchenko on Platform development management (Highload++ 2017):

https://www.youtube.com/watch?list=PLH-XmS0ISi_yn4pCZVOHqWaqQ9trMpyLI&v=J8ZvWzzoVVo



Any Questions?

Join the team!

baschenko@yandex.ru

Skype: baschenkoandrey

<https://www.linkedin.com/in/andreibashchenko/>