Numatica

Scalable Software. Real-time Big Data Analytics.

Big Data, Simple and Fast:

Addressing the Shortcomings of Hadoop

Jacek Kruszelnicki, Numatica Corporation

E-mail: j a c e k@numatica.com (remove spaces)
Phone: 781 756 8064

Presenter

Numatica
Scalable Software. Real-time Big Data Analytics.

Jacek Kruszelnicki
President & CEO, Numatica Corporation

Focus:

- Distributed, High-throughput, Low-latency software.
- Real-time ("Fast") Big Data.

Services:

- IT Strategy and Planning
- Software Architecture & Design
- Proof of Concept, Quick Start programs
- Software Development
- Trainings, Seminars

Jacek:

- 20+ years of shipping enterprise software
- Biased towards simplicity/design elegance. Vendor–neutral.
- Author, mentor & conference speaker
- M.S. Computer Science, focused on business value of IT

Agenda

Numatica
Scalable Software. Real-time Big Data Analytics.

"Any intelligent fool can make things bigger and more complex... It takes a touch of genius - and **a lot of courage** to move in the opposite direction." - E.F. Schumacher

- Hadoop is **not** a universal, or inexpensive, Big Data stack
- Technical requirements for a flexible Big/Fast Data stack
- Solutions thought to be alternatives to Hadoop
- Why In-Memory Data Grids are a good fit for Big/Fast Data
- How Hazelcast meets the Big/Fast Data requirements
- Focus on architectures, but demo/code samples provided

Not on Agenda

Numatica
Scalable Software. Real-time Big Data Analytics.

- Big Data Tutorial
- Hadoop or any other technology tutorial
- In-depth overview of Big Data market
- Big Data Analytics -> Business Insights
- CAP Theorem discussion

Fast Data - Business Drivers

Numatica
Scalable Software. Real-time Big Data Analytics.

Operational Intelligence:

- Analyze stream of business activities and external stimuli on-the fly
- React to them (preferably) instantaneously
- Real time data stream processing is critical, otherwise business value is lost.

Real-time Big (Fast) Data Analytics examples:

- Dynamic pricing (e-commerce)
- High-frequency trading
- Network security threats
- Credit card fraud prevention
- Factory floor data collection, RFID
- Mobile infrastructure, machine to machine (M2M) applications
- Prescriptive or Location-based applications
- Real-time dashboards, alerts, and reports

Big Data – The 3 Vs Re-arranged

Numatica
Scalable Software. Real-time Big Data Analytics.

Common definition:
Data sets too large and complex to process using standard DB tools and data processing applications (in short: "Will not fit into MS Excel")

Variety

- Structured, semi-structured
- At-Rest, In-Motion
- Variety of formats

Velocity

- Millions of events per second
- Need to re-run analytics frequently (Dashboards, etc)
- Frequently changing data (at rest or in-motion)
- Stale data provides little business value
- Also need to do off-line processing (machine learning)

Volume

- Few Googles and Facebooks out there
- Typical data < 100 TB

Big...and Fast

Numatica
Scalable Software. Real-time Big Data Analytics.

Data In-Motion:

- Rapidly changing, massive stream(s) of data (events)
- Multiple sources, formats
- Needs to be processed in-flight
- Events persisted or not, depending on business value

Data At-Rest:

- Data already persisted, change notification
- Multiple formats
- Re-ingest, re-process

Software stacks to support “In-Motion” and “At-Rest” Big Data are needed.

The Hadoop Stack:

- Currently the large scale data analysis tool of choice
- *De facto* standard, almost synonymous with Big Data
- “Nobody ever got fired for using Hadoop”

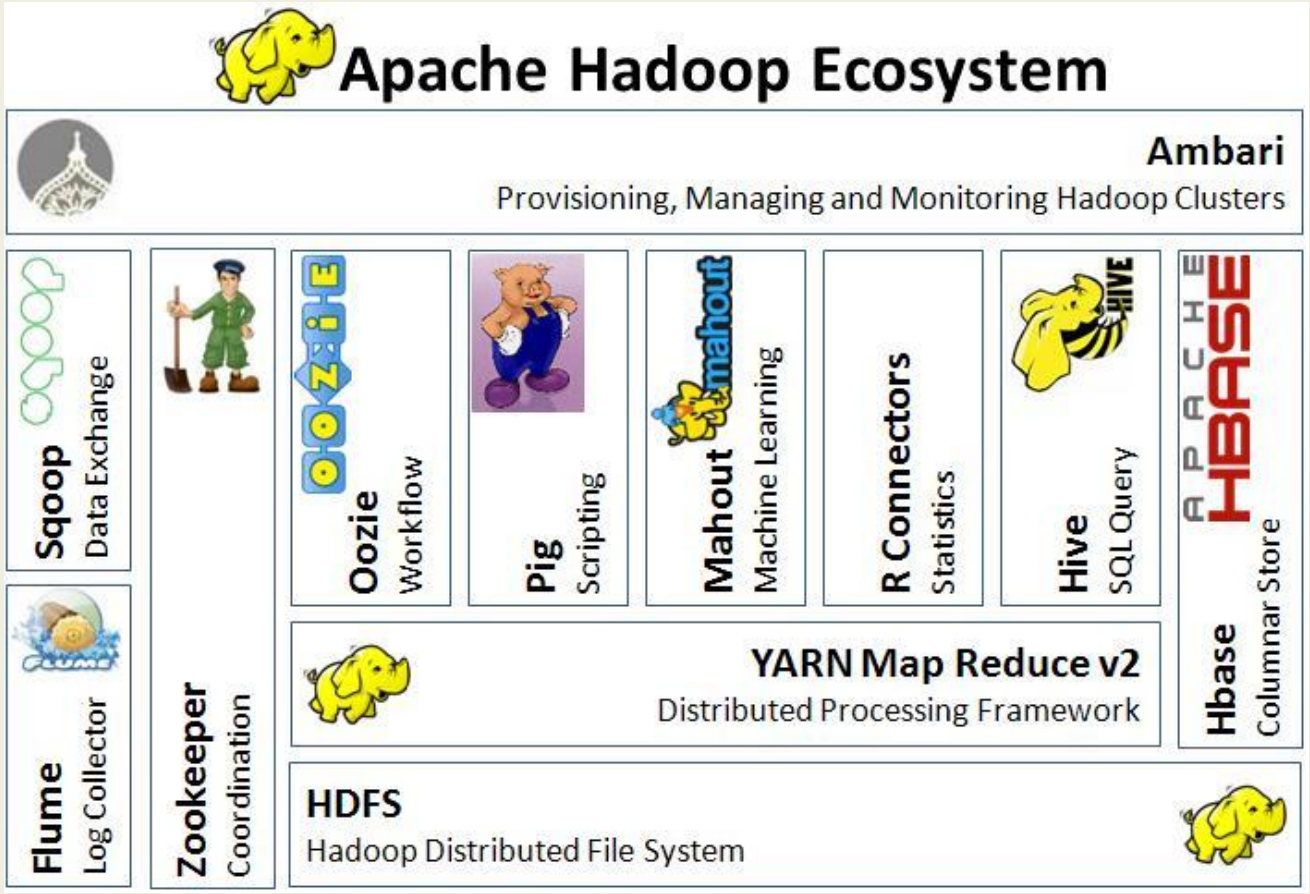
Problems:

- Very complex/expensive to deploy and run -> high TCO
- MapReduce slow, **batch-oriented**, “intrusive”
- No stream processing
- No support for in-memory processing
- Closely coupled with HDFS (third-party solutions of varying quality)
- SQL-on-Hadoop limited and slow

http://hortonworks.com/blog/install-hadoop-windows-hortonworks-data-platform-2-0/
http://www.chrisstucchio.com/blog/2013/hadoop_hatred.html/

Hadoop... and the kitchen sink

Numatica
Scalable Software. Real-time Big Data Analytics.



Not shown:

JobTracker.
TaskTracker,...
Avro
(Serialization)
Chukwa(logs,
incremental)
EMR
BigTop
Spark
Impala (vs Hive)

19 shown, up to 24

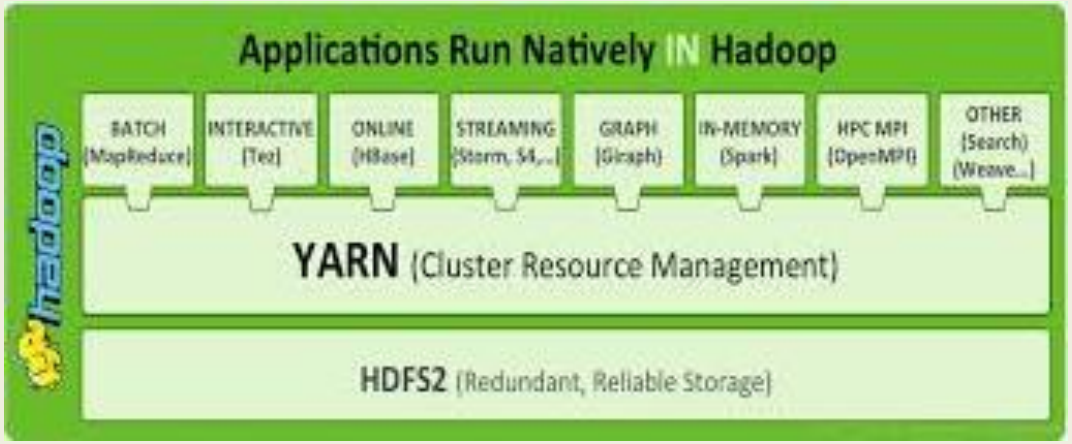
Violates first rule of distributed programming: **DO NOT DISTRIBUTE (unnecessarily)**.

Hadoop 2.0 - High TCO remains

Numatica
Scalable Software. Real-time Big Data Analytics.

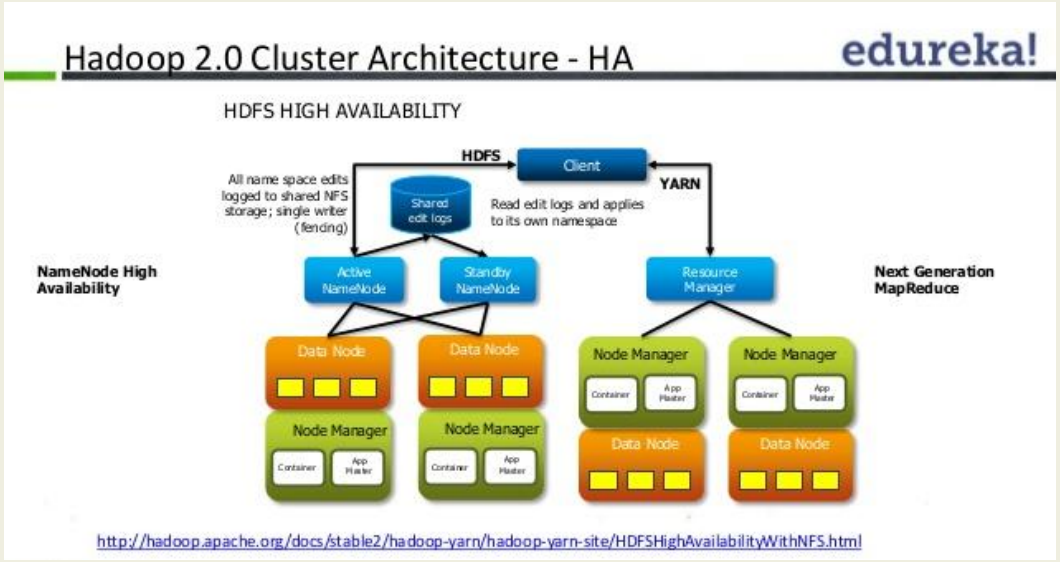
Improved:

- YARN, MapReduce separated
- Removed Name Node/ Job Tracker as SPOF?



Unchanged:

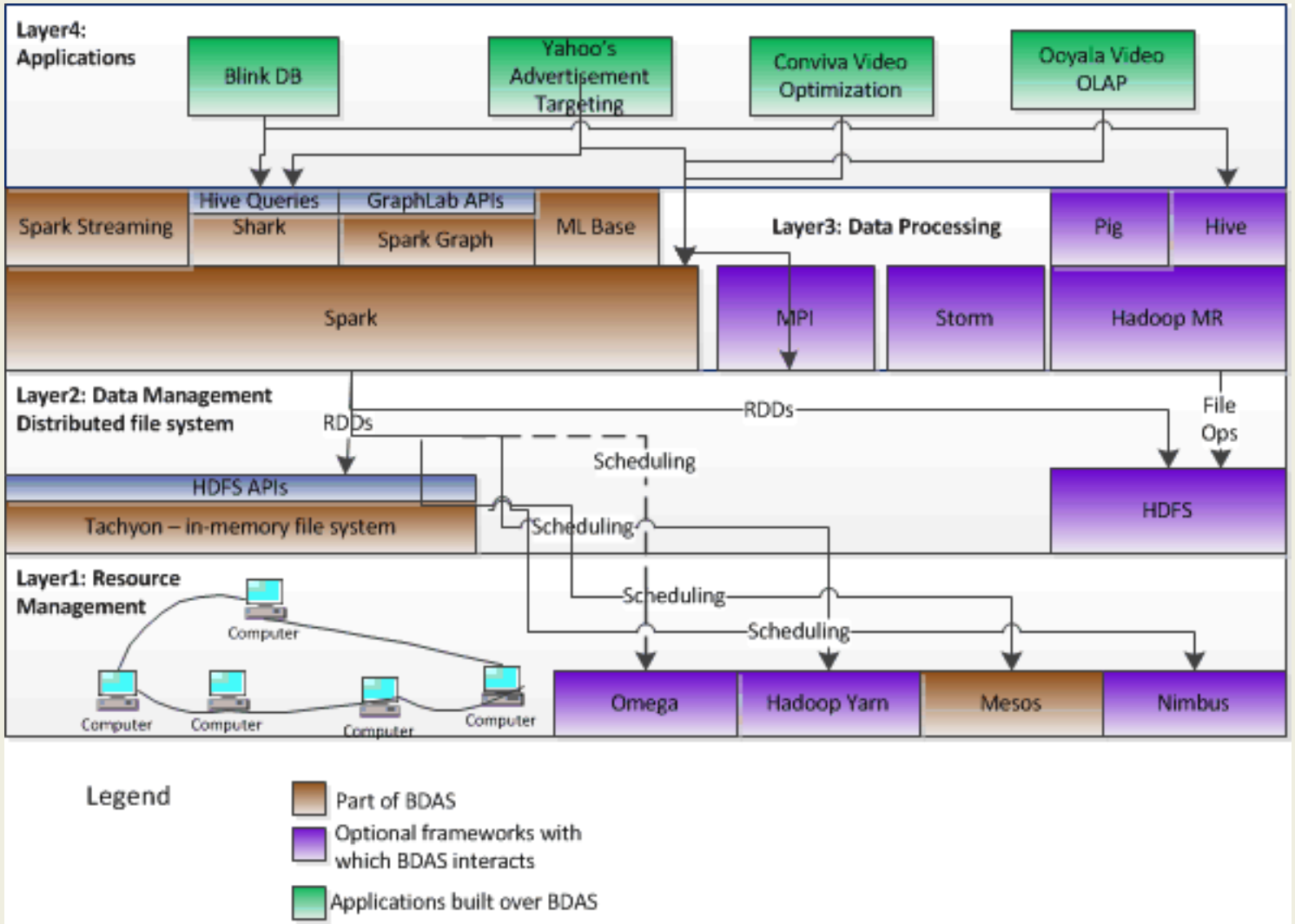
- More complexity
- Low-level abstractions
- Still Master/Slave



Hadoop 2.0 - High TCO remains

Numatica
Scalable Software. Real-time Big Data Analytics.

Hadoop-based stack example from the Web. **Very Complicated:**

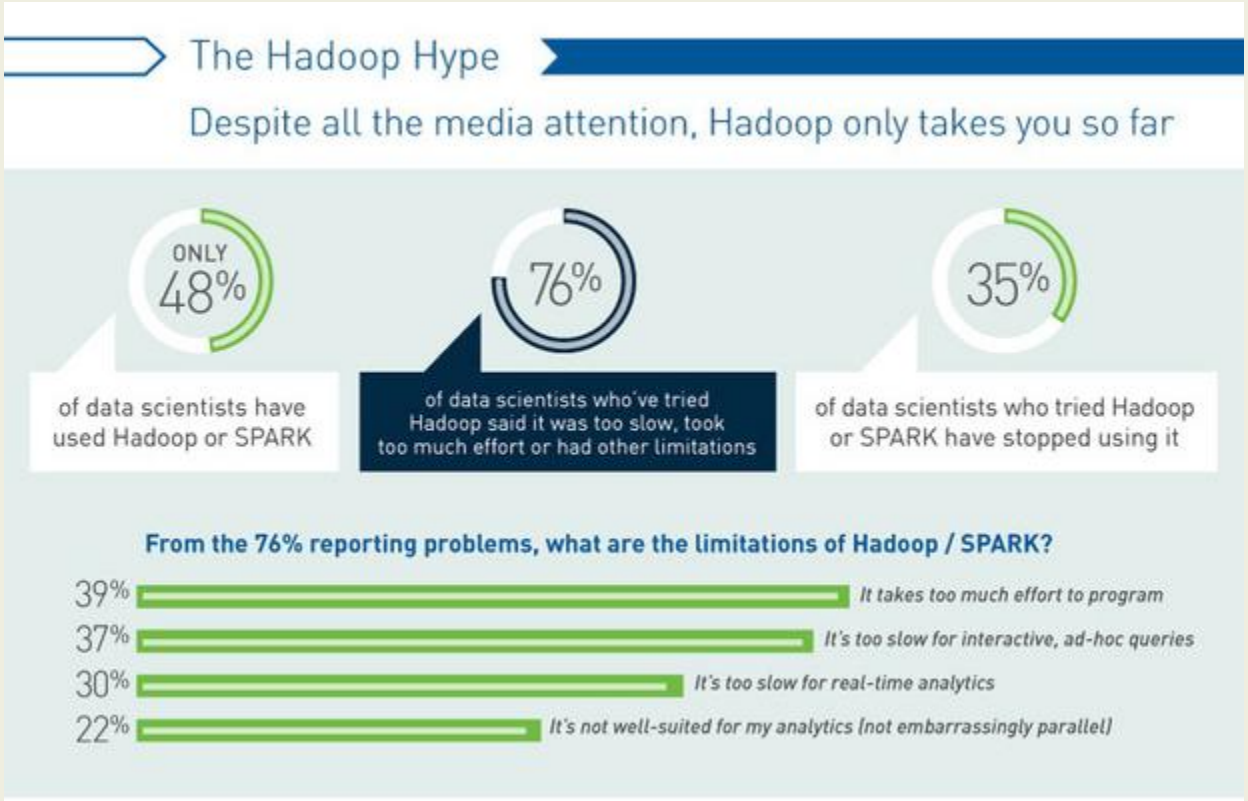


Berkeley Big-data Analytics Stack (BDAS)

Hadoop Disillusionment Phase?

Numatica
Scalable Software. Real-time Big Data Analytics.

Hadoop hyped as a universal, disruptive big data solution



Data Scientists: 76 % felt Hadoop is too slow, too much effort to program

http://www.cio.com/article/2449814/big-data/data-scientists-frustrated-by-data-variety-find-hadoop-limiting.html

MR is on its way out at Google.

http://www.datacenterknowledge.com/archives/2014/06/25/google-dumps-mapreduce-favor-new-hyper-scale-analytics-system/

Some Hadoop Alternatives

Numatica
Scalable Software. Real-time Big Data Analytics.

Hadoop "extensions"

- Spark
- Shark (Spark on Hive)
- Storm
- Kafka

Still needs many of Hadoop modules: HDFS, YARN, Pig, Hive, HBase, JobTracker.TaskTracker,...

MPP DBs

- Teradata
- Vertica
- Greenplum

Proprietary, complex, VERY expensive, query language not Turing complete

NoSQL DBs

Column: Cassandra, Hbase, etc.
Document/K-V: MongoDB, CouchDB, Riak, etc.
In-Memory: VoltDB, etc.

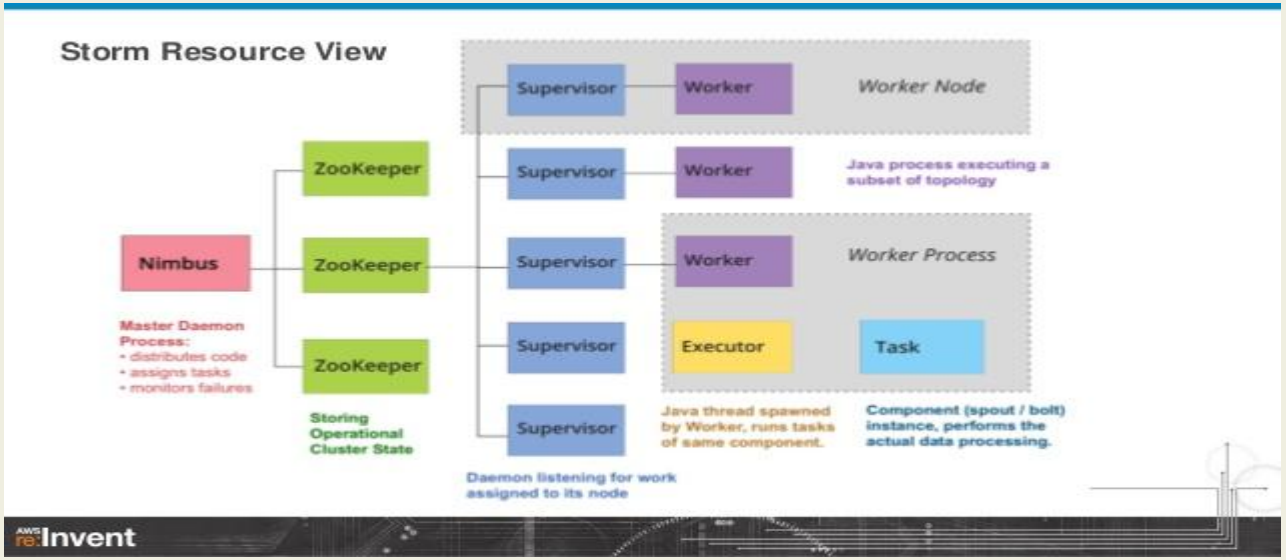
No ACID/Referential integrity,
No triggers, foreign keys
Mongo/Couch ->JSON-centered, Cassandra - complex
Query language proprietary, subset of SQL
VoltDB – precoded stored procs, no ad-hoc
Not Turing complete

In-Memory Data Grids

- Coherence (commercial
- Hazelcast (OSS)
- Terracotta (commercial)
- Gemfire (commercial)
- GridGain (OSS/commercial)
- Gigaspaces XAP (OSS)

Storm/Kafka

- **Intrusive** (introduces exotic abstractions): Streams, Spouts, Bolts, Tasks, Workers, Stream Groups, Topologies
- **Low-level:** Shell scripting, Clojure code here and there
- **Complex Admin:** Everything requires ZooKeeper, Nimbus is a SPOF

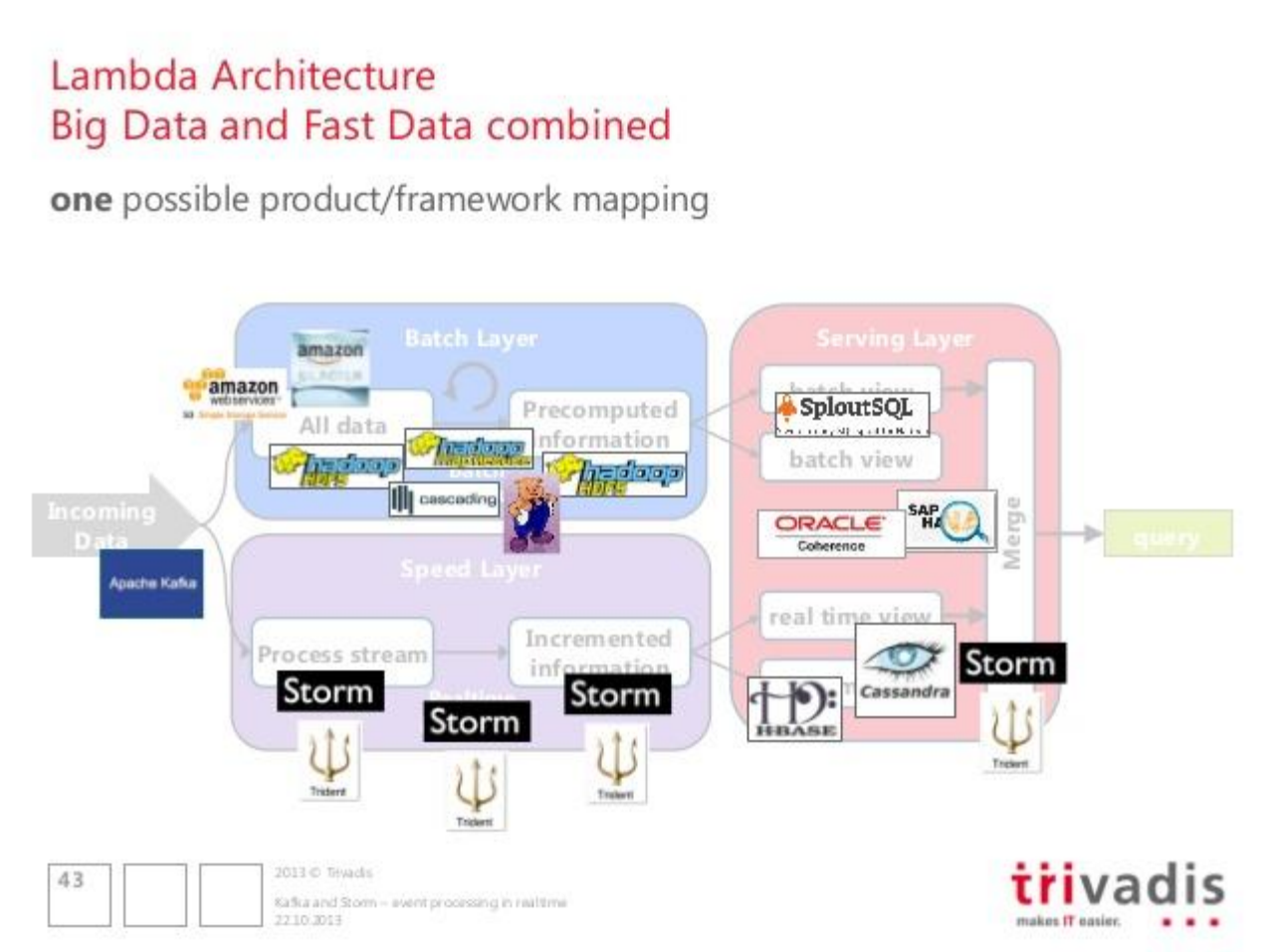


https://news.ycombinator.com/item?id=8416455

Big+Fast Alternatives part 2

Numatica
Scalable Software. Real-time Big Data Analytics.

Lambda Architecture example from the Web. Lots of moving parts:



Big, Fast Stack - Technical Requirements

Numatica
Scalable Software. Real-time Big Data Analytics.

Successful Big + Fast Data stack should have:

- Architectural **simplicity** and elegance [lowers Total Cost of Ownership (TCO)]
- **Low** Transactional and Data **Latency** [response <1s, “fresh” or real-time data]
- High-throughput, overall up- and out- Scalability
- High-throughput Data Stream/Complex Event Processing
- SQL-like querying, ACID, Txs (including XA)
- Distributed Execution Framework
- High Availability and Fault Tolerance.
- Stateless, decentralized, elastic cluster management

In-Memory Grids for Big And Fast Data

Numatica
Scalable Software. Real-time Big Data Analytics.

- Databases (RDBMS, NoSQL) are not enough
  - SQL or SQL-like languages are not Turing complete
  - Object-oriented/functional/parallel programming abstractions needed
- Technology moving forward (6x sequential, 100K x random):
  - RAM is the new disk (ns)
  - DISK is the new tape (ms)
- In-memory data is volatile/perishable?
  - In-memory data can be persisted, if so desired
  - No need to achieve archival durability.
    - Most Big Data brought in already stored somewhere else
    - Stream data may not be relevant for long, typically limited retention time

Why Hazelcast?

Numatica
Scalable Software. Real-time Big Data Analytics.

- **The most intellectually elegant distributed in-memory data/compute grid.**
- Minimalism as design aesthetic:
  Non-intrusive,
  No dependencies
  3.1MB single jar library.
- Implements Java APIs (Map, List, Set, Queue, Lock) in a distributed manner
- Distributed Execution Framework (extension of Java's Executor Service)
  Distributed Queries (SQL/Predicate), Data affinity (execution on specific node execution)
- Peer-peer architecture, no single point of failure
- Elastic cluster management Java API included
  Auto-discovery of nodes and re-balancing
- Apache License 2, commercial extensions + support

Hazelcast vs Hadoop

Numatica
Scalable Software. Real-time Big Data Analytics.

Java8 + Hazelcast (single 3.1 MB jar)	Java + Hadoop
Pluggable persistence (HDFS, MapR, RDBMS)	HDFS
MapReduce	MapReduce
Data manipulation & querying	Hive (not OLTP)
In-memory parallel processing	Spark
Stream processing	Storm
Messaging	Kafka
Scalable and Elastic	ZooKeeper
Cluster Management	ZooKeeper, YARN, Mesos
Elastic, simple (automatic cluster re-balancing)	N/A

Considerations

Numatica
Scalable Software. Real-time Big Data Analytics.

- RAM currently maxes out at ~ 640GB/server (256GB?)
- RAM still more expensive than SSDs and HDDs
- Garbage collection

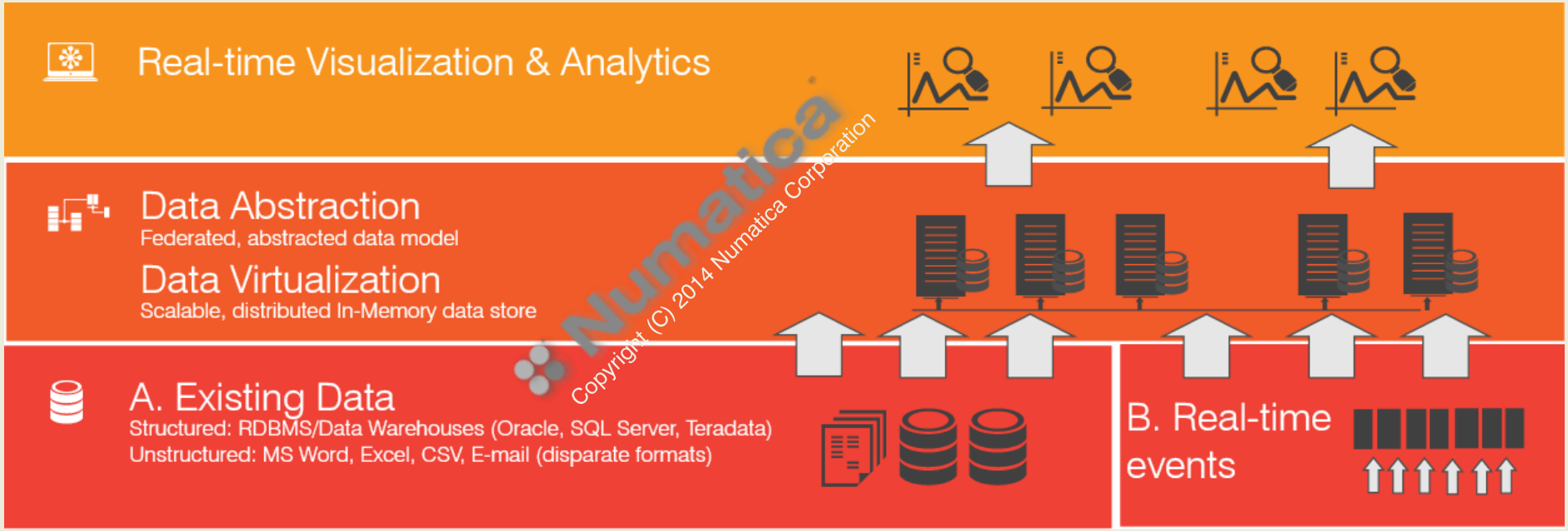
- Cost and capacity limitations will disappear over time
- Off-heap memory and specialized JVMs (Azul, etc.)

From Technologies to Platforms

Numatica
Scalable Software. Real-time Big Data Analytics.

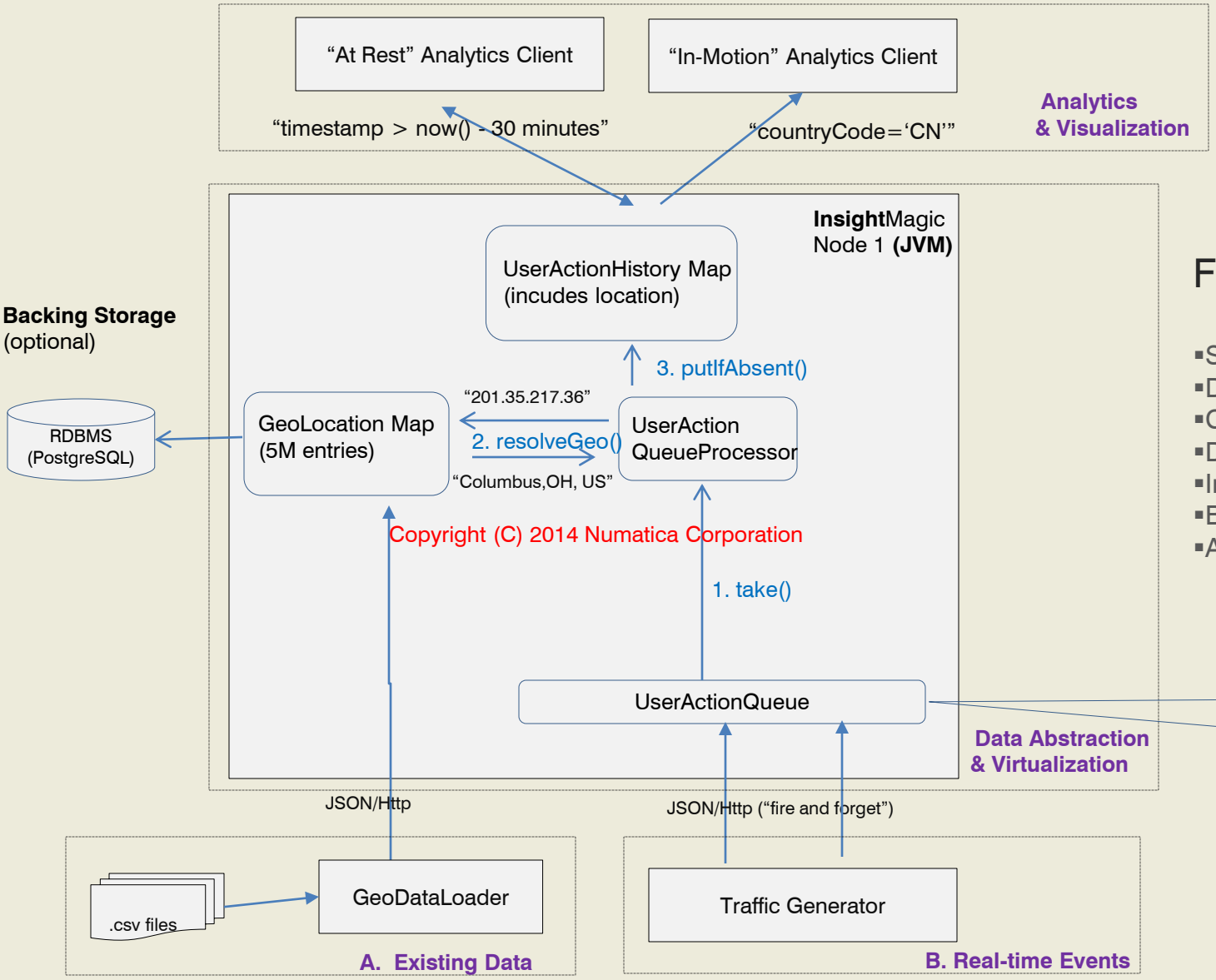
In-Memory Data/Compute Grids not enough:

- Backing Storage: RDBMS, NoSQL, HDFS GlusterFs, XFS
- Enterprise Message Store(s)
- Multi-Source Data Harvesting, Ingestion, Transformation
- Data Abstraction, Modeling, Querying, Visualization



The Demo: Clickstream Analytics + Intrusion Detection

Numatica
Scalable Software. Real-time Big Data Analytics.



Features:

- SOA Architecture
- Data Ingestion
- Complex Event Processing
- Dimension, Fact Maps
- In-Motion, At-Rest Analytics
- Backing Storage
- ACID + Txs available

Q & A

Numatica
Scalable Software. Real-time Big Data Analytics.

“Intellectuals solve problems. Geniuses prevent them.”

-- Albert Einstein

More questions? Feel free to contact me at:

Jacek Kruszelnicki, Numatica Corporation

E-mail: j a c e k@numatica.com (remove spaces)
Phone: 781 756 8064

Appendix

Numbers Everyone Should Know*

Operation	Time (nsec)	Time
L1 cache reference	0.5	0.5s
Branch mispredict	5	5s
L2 cache reference	7	7s
Mutex lock/unlock	25	25s
Main memory reference	100	1m40s
Send 2K over 1Gbps network	20,000	5h30m
Read 1MB sequentially from memory	250,000	~3days
Disk seek	10,000,000	~6days
Read 1MB sequentially from disk	20,000,000	8months
Send packet CA -> NL -> CA	150,000,000	4.75years

* Jeff Dean. Designs, lessons and advice from building large distributed systems. Keynote from LADIS, 2009.