In-Stream Processing Service Blueprint

Reference architecture for real-time Big Data applications

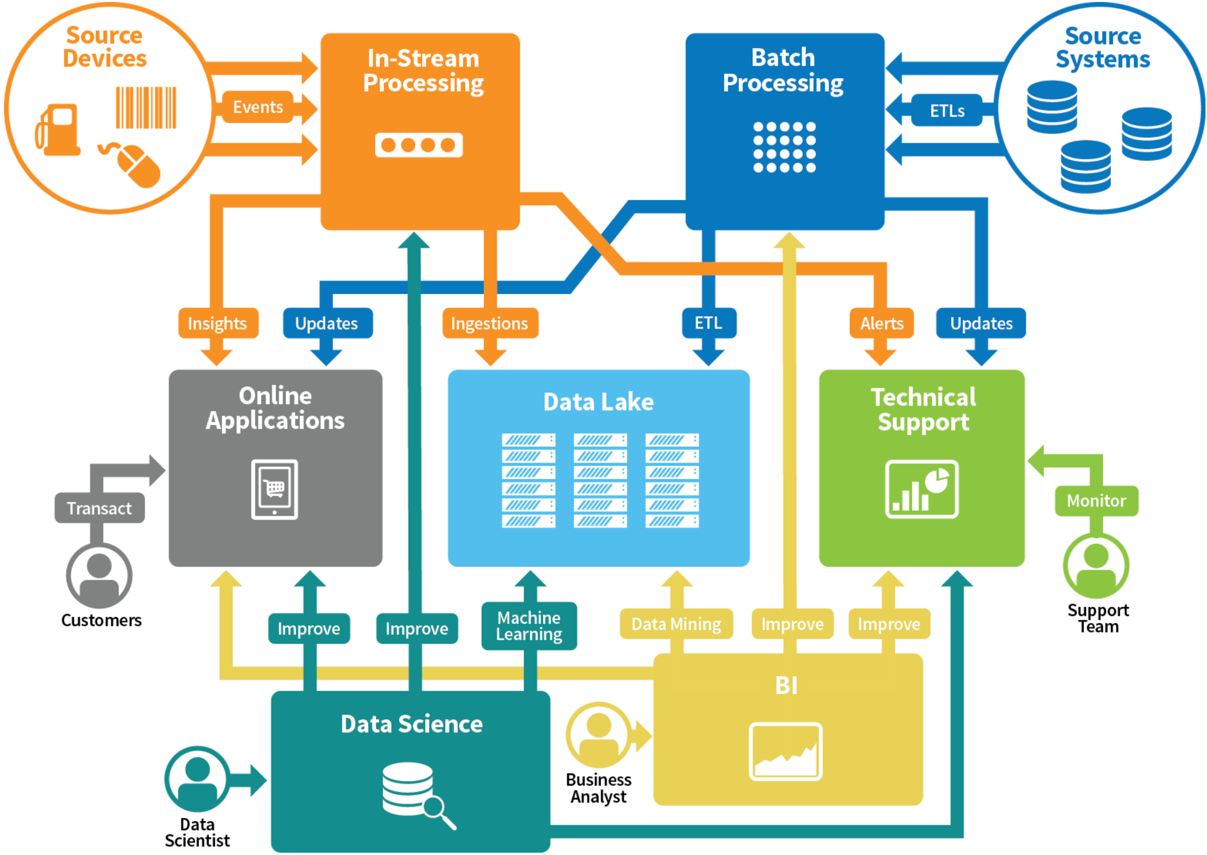
Anton Ovchinnikov, Data Scientist, Grid Dynamics

2016
CEE-SEC(R)

What we'll talk about today

- What's In-Stream Processing?
- How it's used to process huge data streams in real time
- How to build in-stream processing with open source
- All about scale and reliability considerations
- Example of large-scale customer implementation
- Where can you learn more (hint: blog.griddynamics.com)

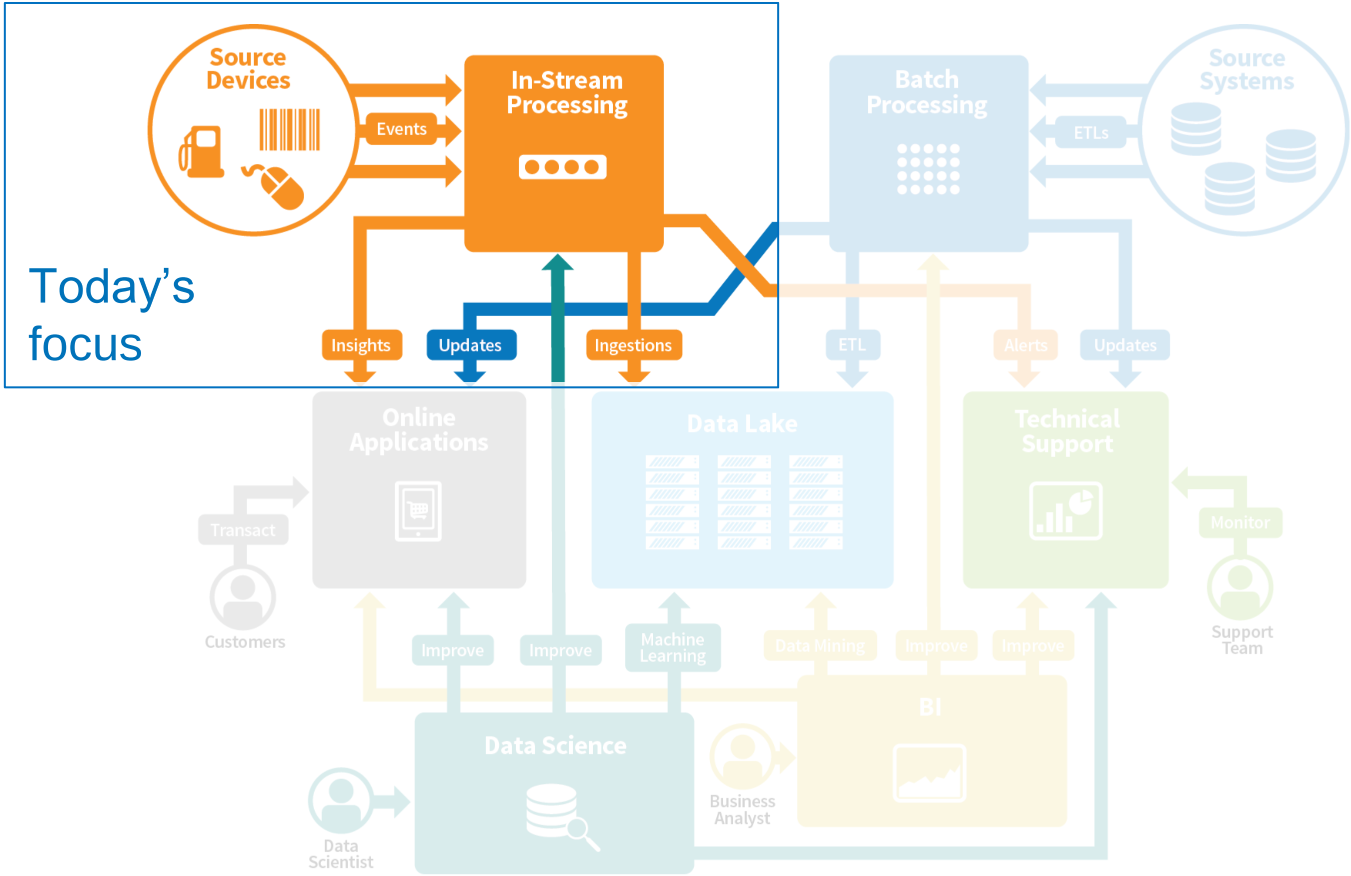
In a complex landscape of Big Data systems…



Grid Dynamics

…in-stream processing service is

an approach to build real-time Big Data applications



Grid Dynamics

Multiple industries and use cases

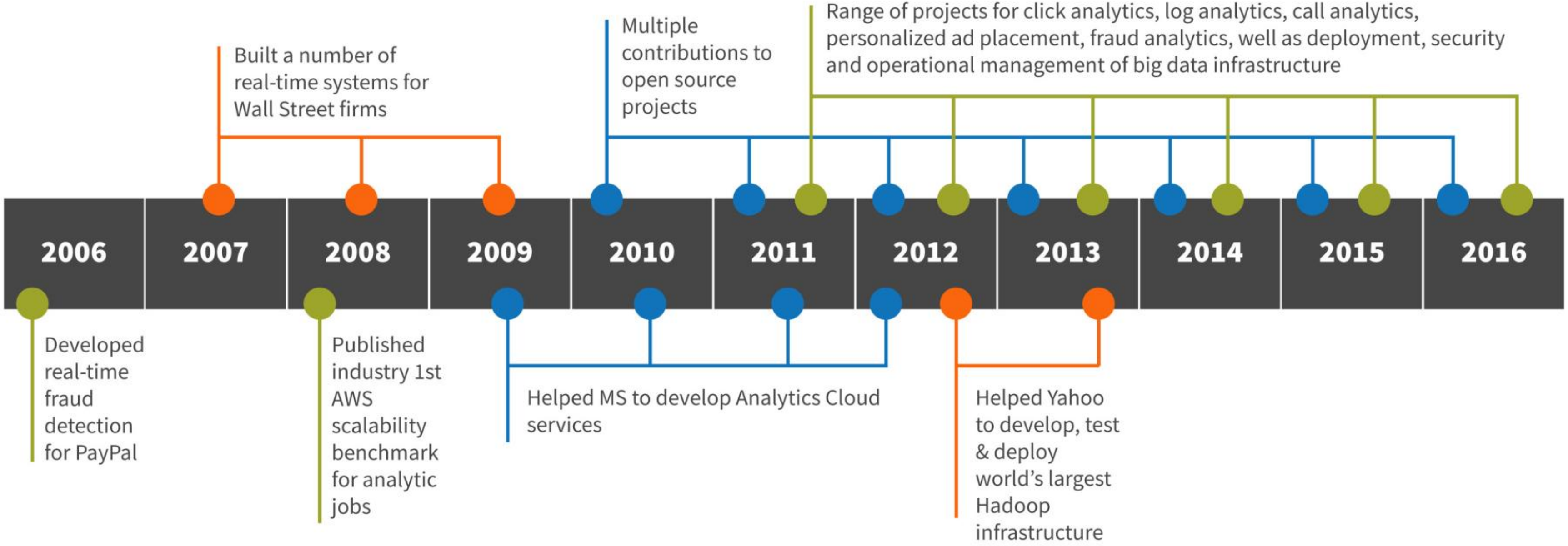
- Fraud detection
- Sentiment analytics
- Preventive maintenance
- Facilities optimization
- Network monitoring
- Intelligence and surveillance
- Risk management
- E-commerce

- Clickstream analytics
- Dynamic pricing
- Supply chain optimization
- Predictive medicine
- Transaction cost analysis
- Market data management
- Algorithmic trading
- Data warehouse augmentation

Open source world is diverse and confusing

What credentials do we have to talk about this?

Big Data history @Grid Dynamics



Built a number of real-time systems for Wall Street firms

Multiple contributions to open source projects

Range of projects for click analytics, log analytics, call analytics, personalized ad placement, fraud analytics, well as deployment, security and operational management of big data infrastructure

Developed real-time fraud detection for PayPal

Published industry 1st AWS scalability benchmark for analytic jobs

Helped MS to develop Analytics Cloud services

Helped Yahoo to develop, test & deploy world's largest Hadoop infrastructure

Blueprint goals

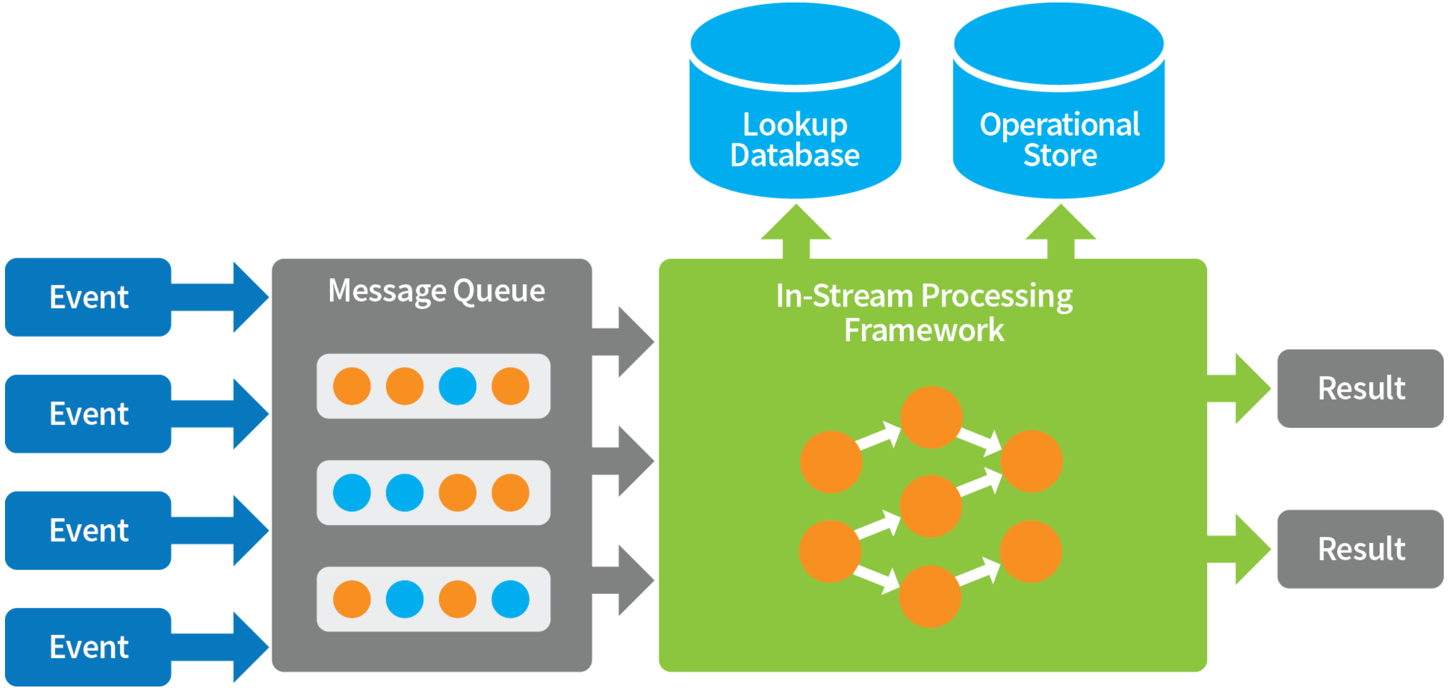
Pre-integrated	Cloud-ready	Production-ready	Enterprise-grade
Built 100% from leading open source projects	Portable across clouds	Proven mission-critical use	Extendable

Grid Dynamics

Target performance & reliability SLAs

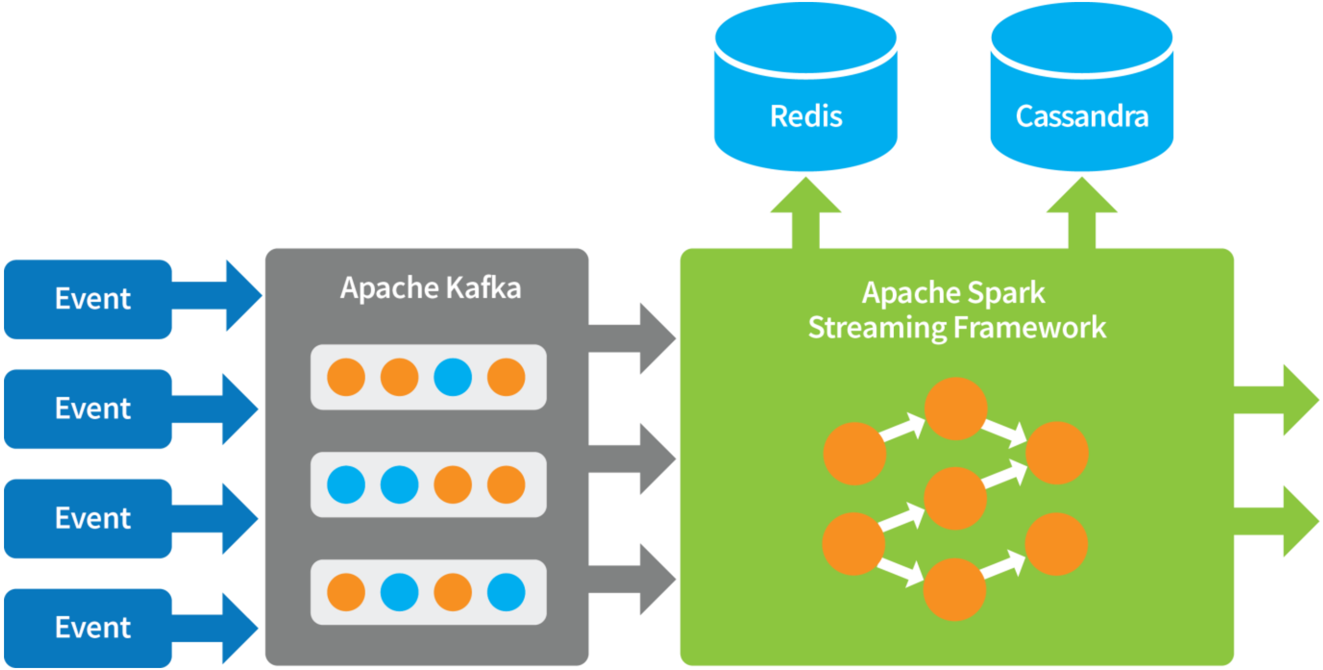
Throughput	Up to 100,000 events per second
Latency	1-60 seconds
Retention	Raw data and results archived for 30 days
Reliability	Built-in data loss mitigation mechanism in case of faults
Availability	99.999 on commodity cloud infrastructure

Conceptual architecture



Grid Dynamics

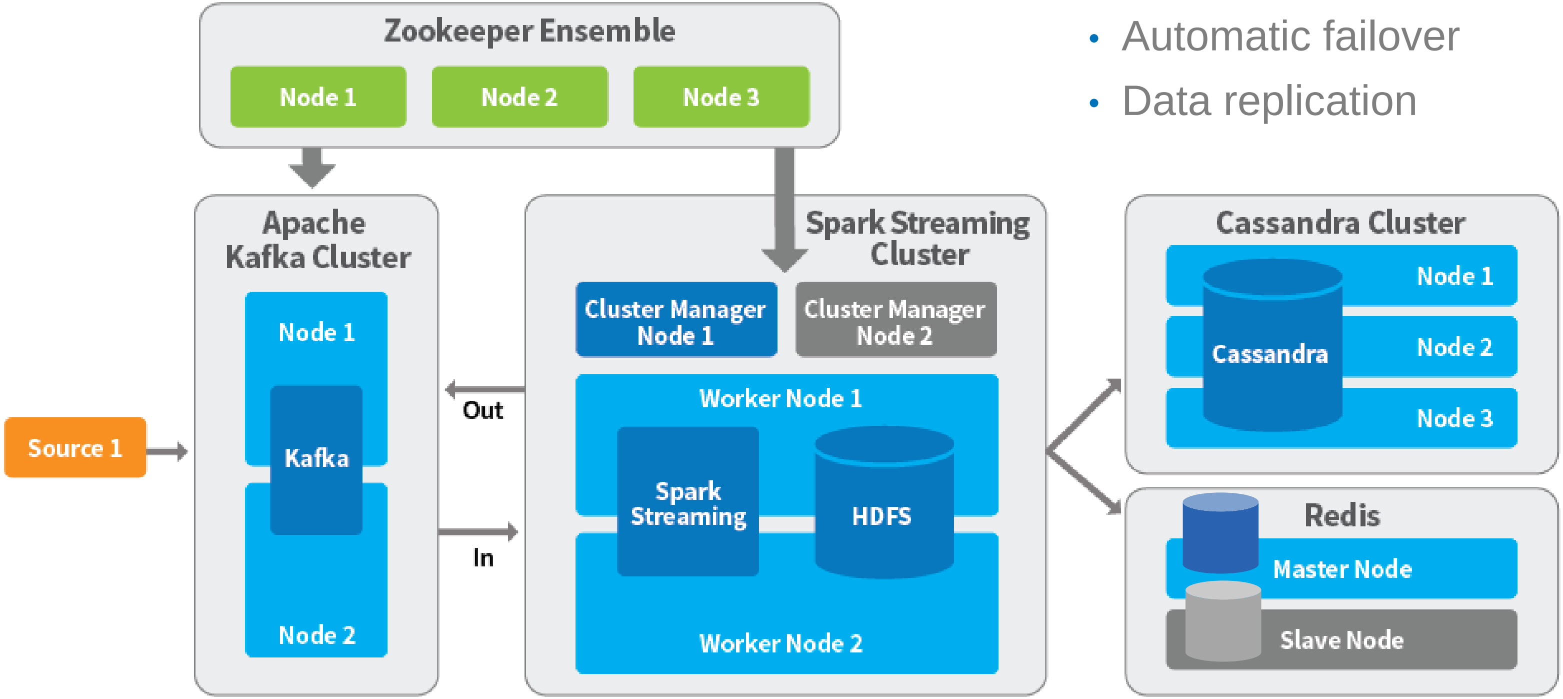
Selected stack



- REST API
- Message Queue
- HDFS

Every component is scalable in its own way

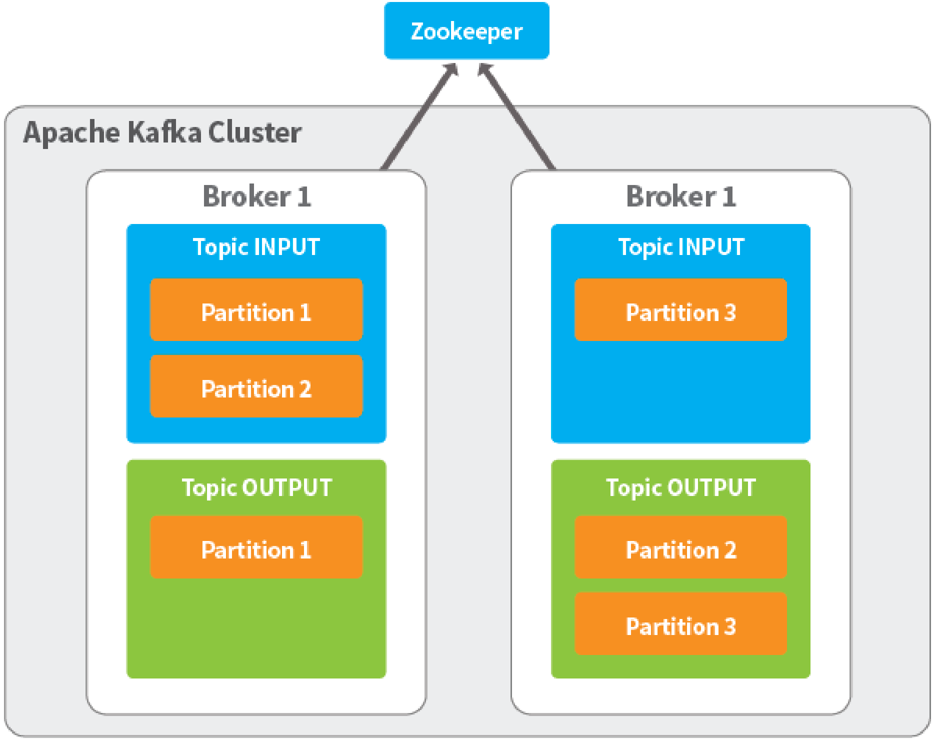
- No single point of failure
- Automatic failover
- Data replication



Grid Dynamics

Multi-node Kafka cluster

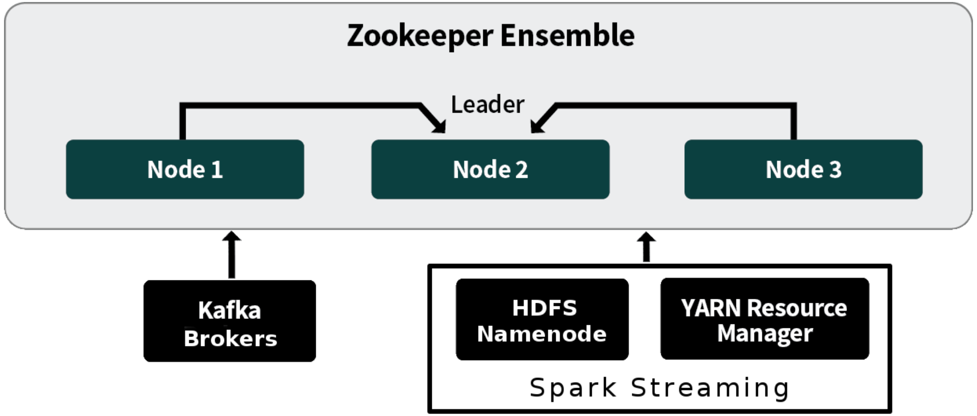
- Undisputed modern choice for real time MOM
- Retention and replay
- Scalable via partitioning
- Persistent
- Super-fast



Grid Dynamics

Single Zookeeper cluster for all components

- Distributed coordination service, facilitates HA of other clustered services
- Guaranteed consistent storage
- Client monitoring
- Leader election



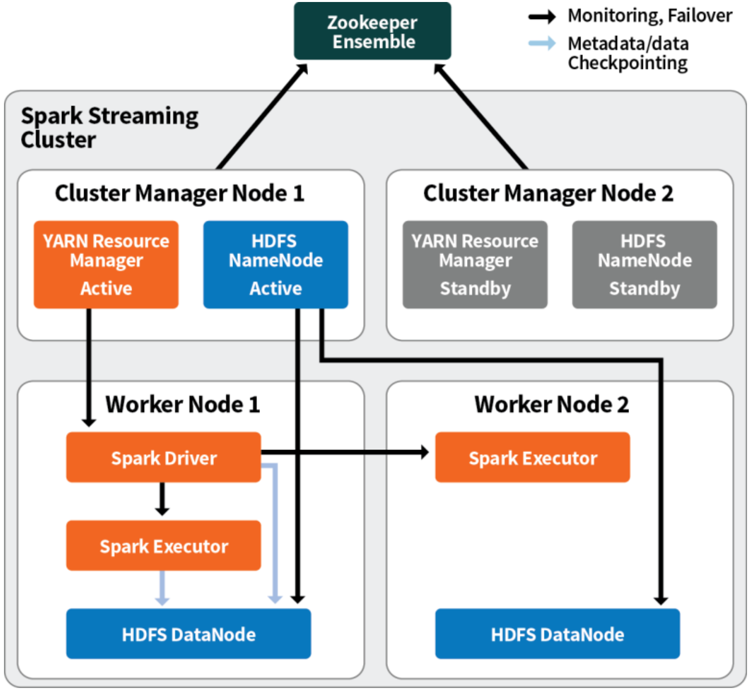
Spark streaming:
Central component of the platform

Why Spark streaming?

- Leading In-Stream Processing middleware
- Active community
- Rate of adoption
- Vendor support
- Excellent integration with Hadoop eco-system

Key architectural considerations

- Runs on top of Hadoop
- Out-of-the-box integration with Kafka
- Support for machine learning pipelines

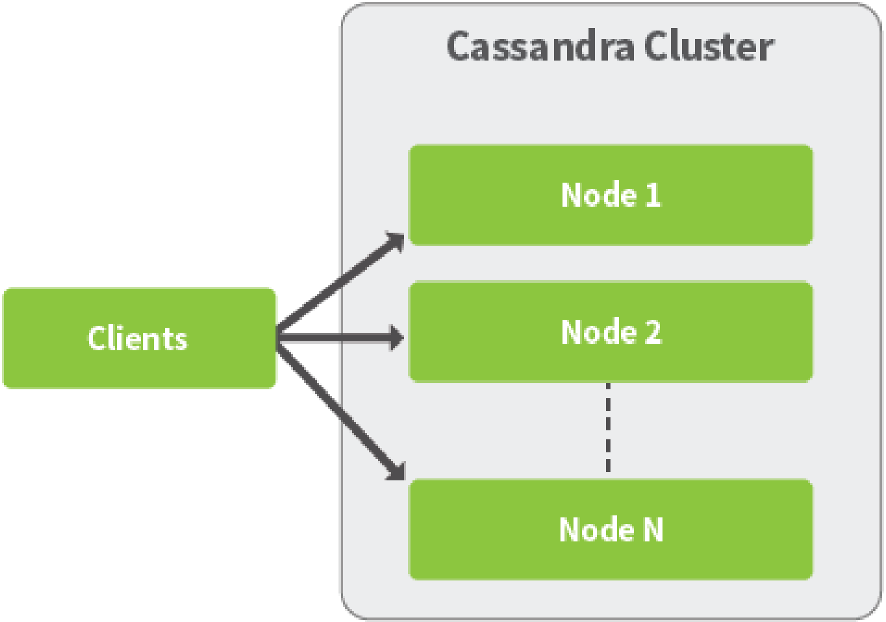


Grid Dynamics

Cassandra as operational store

- Massively scalable, highly available NoSQL database
- Ideal choice as large operational store (100s of GB) for streaming applications
- Needed when event processing is stateful, and the state is quite large

Example: stores user profiles as they are being updated in real time from clickstreams

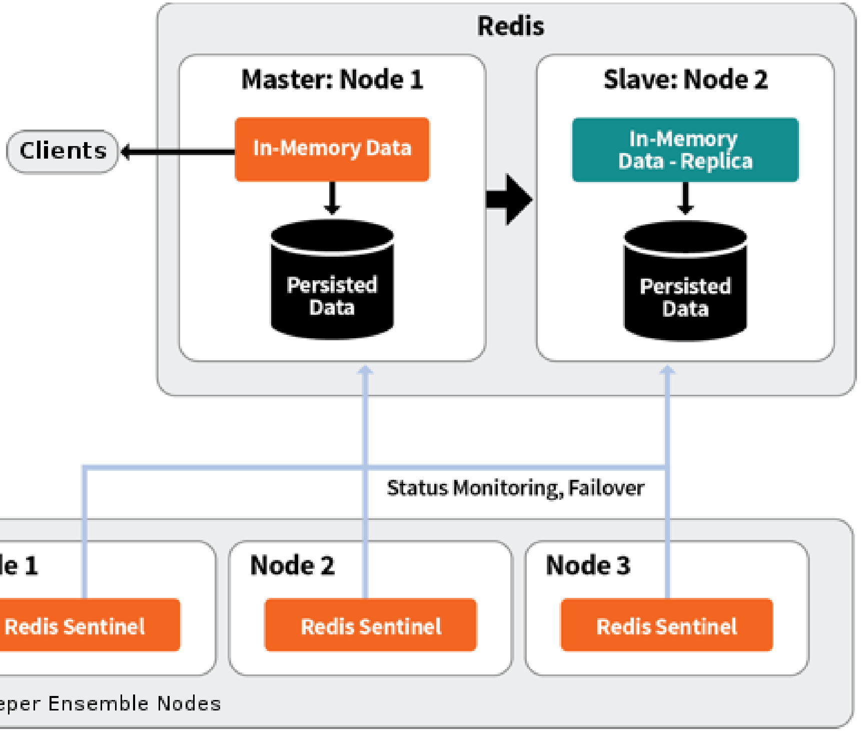


Grid Dynamics

REDIS as a lookup database

- Simple, cheap and super-performant lookup store
- Needed when event processing requires frequent access to GBs of reference data
- Can be updated from outside
- Master/slave architecture

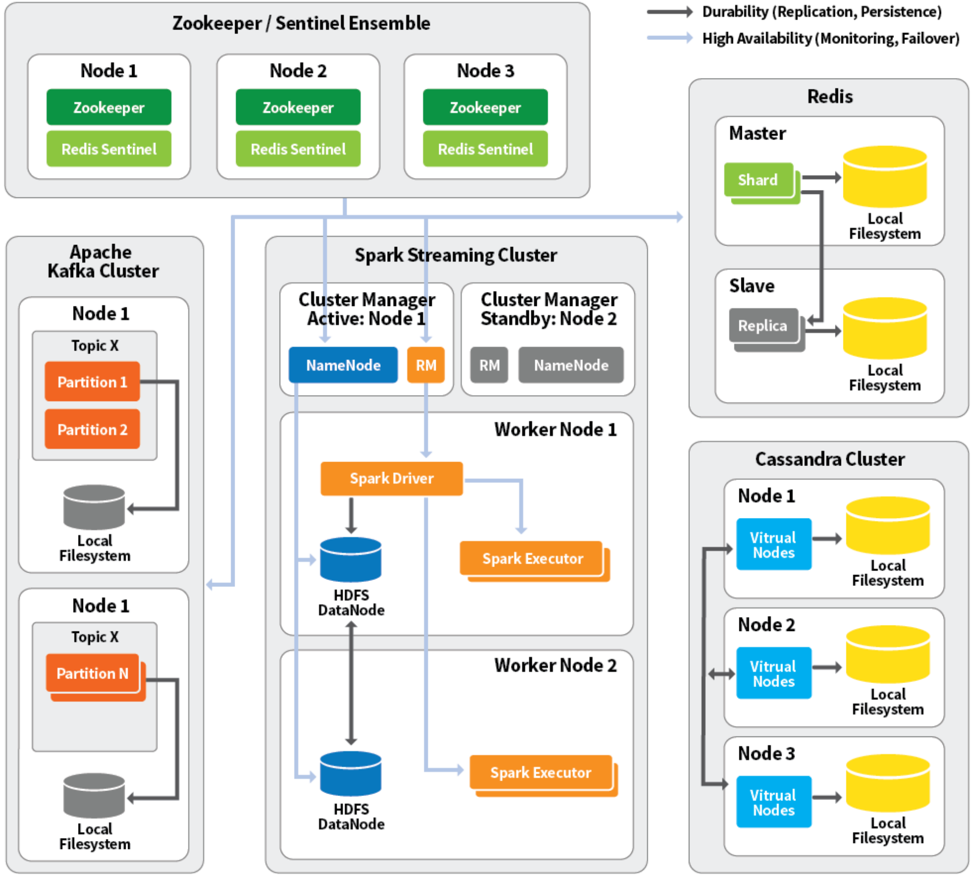
Example: IP geo-mapping, dictionaries, training sets



Putting all the pieces together:
end-to-end platform configuration

- No single points of failure
- No bottlenecks
- Scaling or recovering any component cluster mitigates availability issues

- Caveat: pathologies do happen, even in this design – for example, dynamic repartitioning is not supported



Case study:

Large media agency

Business opportunity

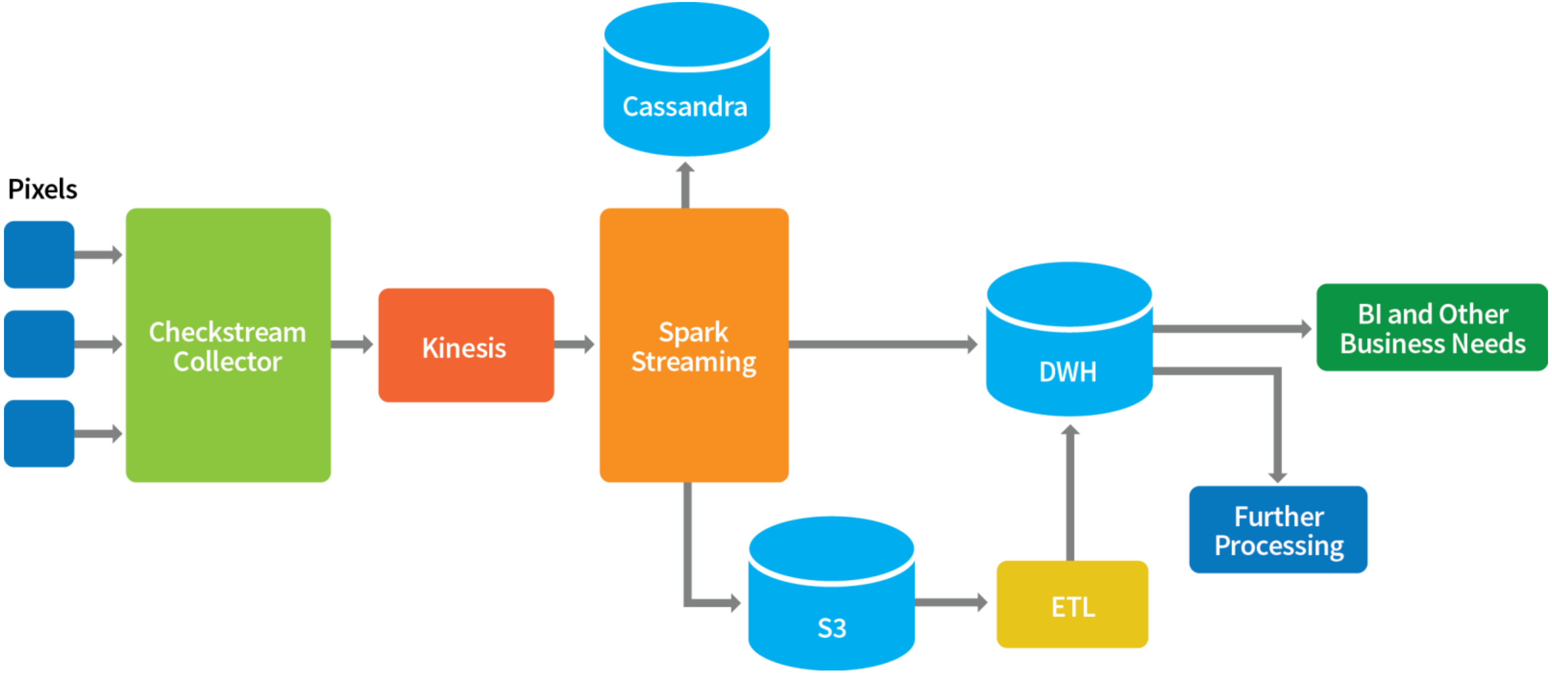
- Real-time popularity trends for the content across all client’s properties drives audience coverage with the most interesting, trending articles

Work done

- Implementation used Amazon Kinesis & Amazon EMR/Spark Streaming stack
- Data egress: S3 and Amazon Redshift

Case study:

Implementation details



Summary

- In-Stream Processing is a hot new technology
- It can process mind-boggling volumes of events in real-time and discover insights
- You can build a whole platform with 100% open source components
- We give you a complete blueprint on how to put it together
- It will run on any public cloud

What we didn’t get to talk about today

- Docker, Docker, Docker: how to make auto-deployment and auto-scaling work
- Data scientist’s kitchen: what they are doing when no one is watching
- Cloud sandbox for our In-Stream Processing Blueprint: how to take it for a spin on AWS
- Demo app: see how social analytics is done using the blueprint

- All this, and more: coming up soon in our blog (blog.griddynamics.com)

- Please, subscribe!

Thank you!

Anton Ovchinnikov: aovchinnikov@griddynamics.com

Grid Dynamics blog: blog.griddynamics.com

Follow up on twitter: @griddynamics

We are hiring!

https://www.griddynamics.com/careers

Grid Dynamics

Grid Dynamics