Monitoring WLCG with the *lambda* architecture

A new processing framework based on Hadoop (and friends)

Luca Magnoni
CERN IT

Outline

- Experiment Dashboard as use case
- WLCG Monitoring: challenges
- A new monitoring architecture
- Hadoop at work:
  - Data representation: make you choices
  - Performance results
- Real-Time layer: options and tools

Experiment Dashboard (ED)

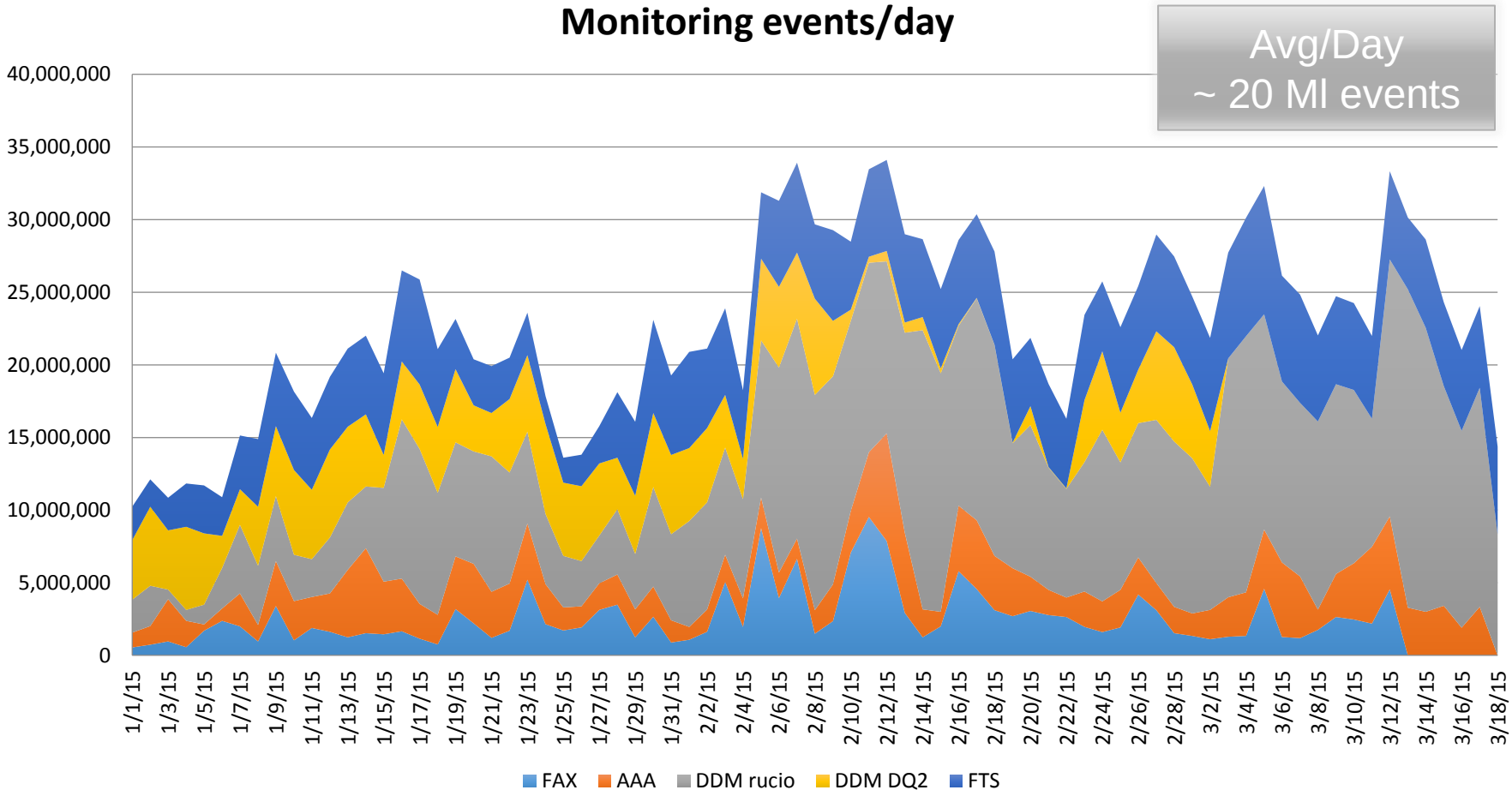
Monitoring
WLCG computing activities since 2006

- A common monitoring framework developed at CERN
- Multiple Targets:
  - Data activities
  - Job Processing
  - Services and Sites Availability
- Different Perspectives:
  - Users, Experts, Sites

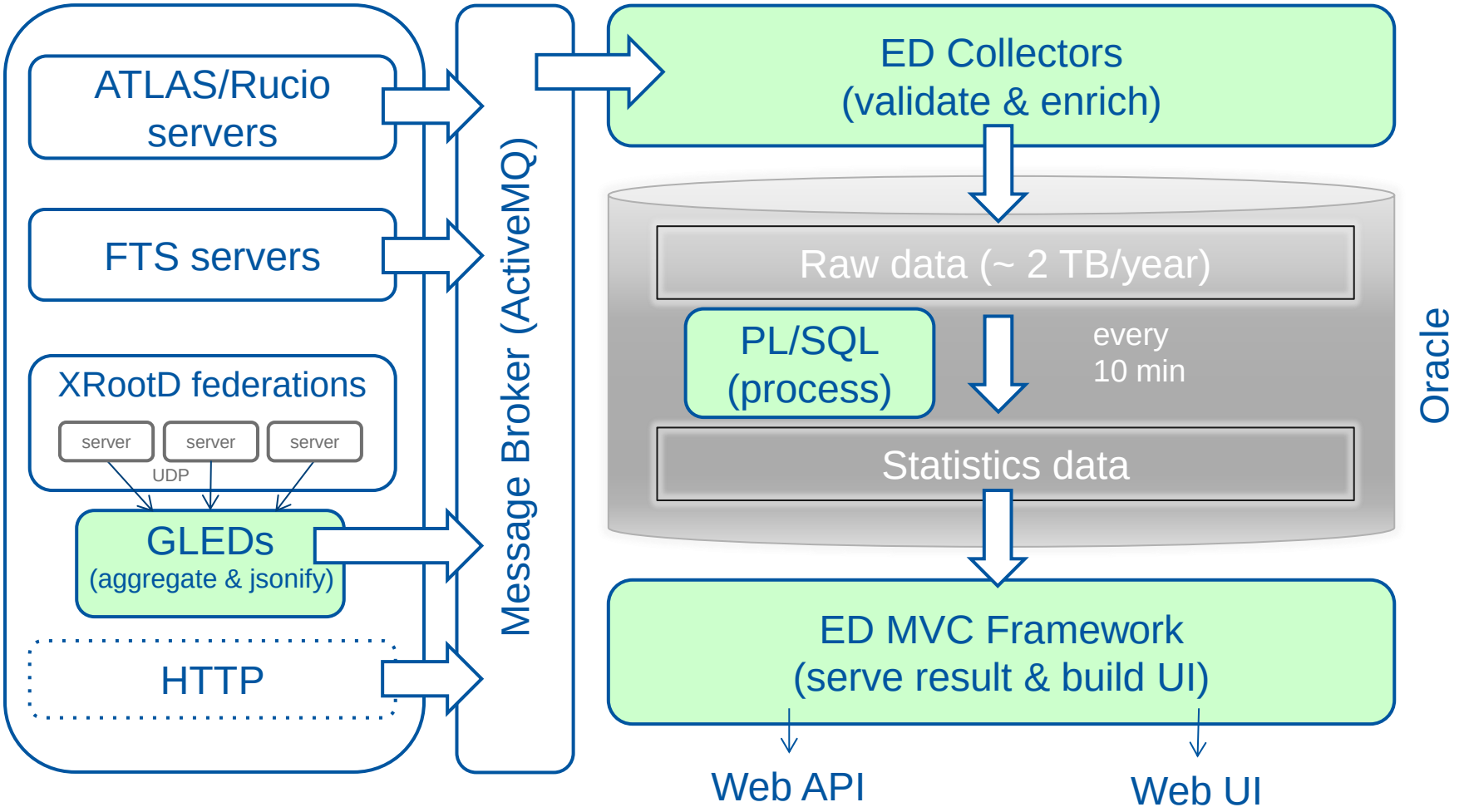
What does it do? *Analytics.*

Collects and processes monitoring data to build custom visualization

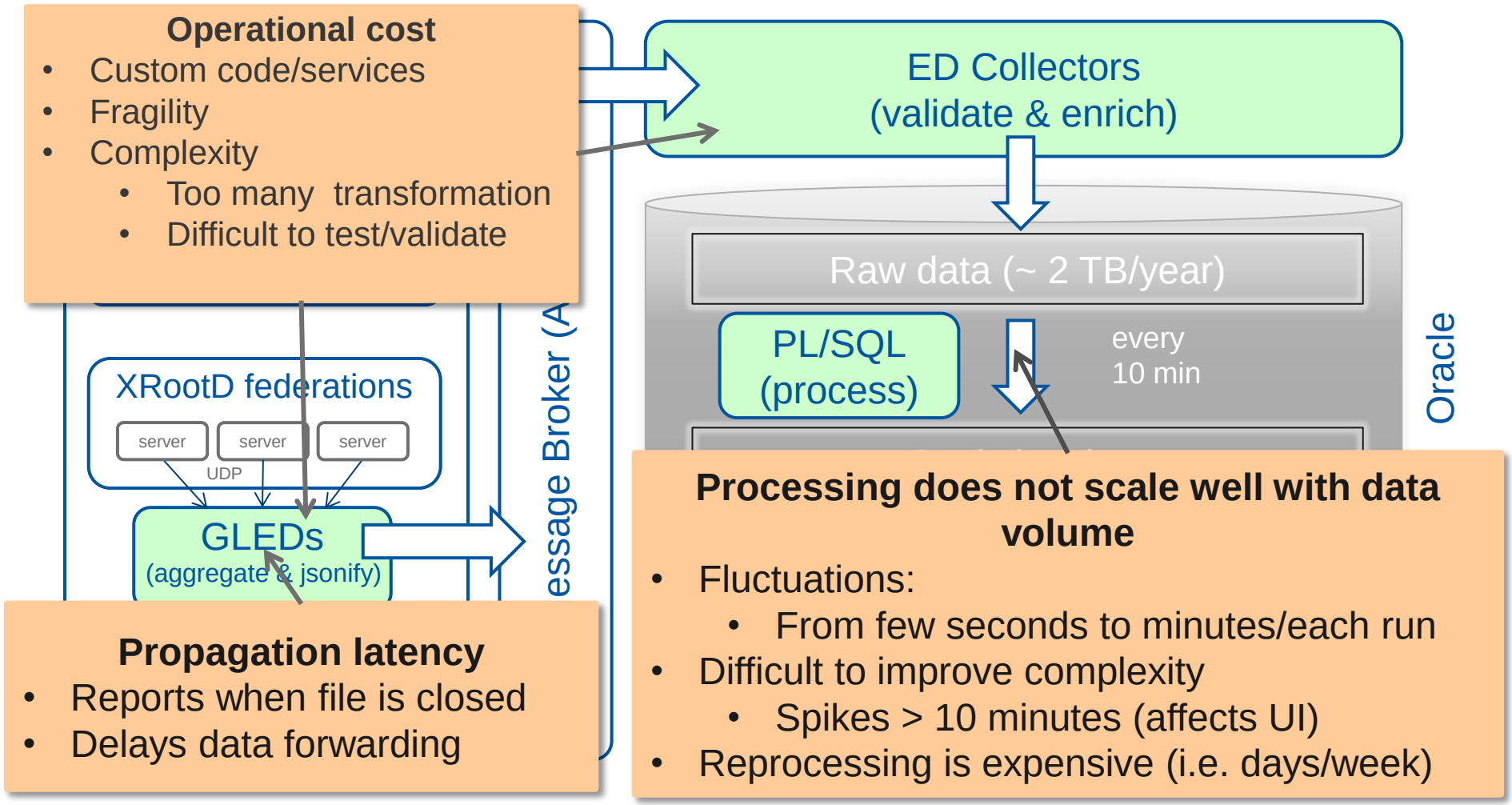
Monitoring WLCG Data Activities



Current ED Data Pipeline



It works, but…


Operational cost
• Custom code/services
• Fragility
• Complexity
• Too many transformation
• Difficult to test/validate
ED Collectors
(validate & enrich)
Raw data (~ 2 TB/year)
PL/SQL
(process)
every
10 min
Oracle
XRootD federations
server
server
server
UDP
GLEDs
(aggregate & jsonify)
essage Broker (A
Propagation latency
• Reports when file is closed
• Delays data forwarding
Processing does not scale well with data volume
• Fluctuations:
• From few seconds to minutes/each run
• Difficult to improve complexity
• Spikes > 10 minutes (affects UI)
• Reprocessing is expensive (i.e. days/week)

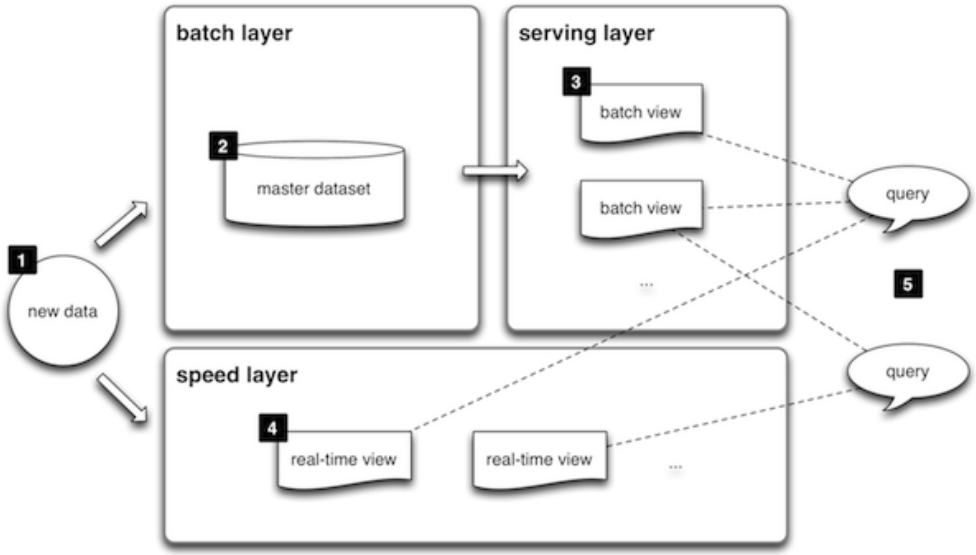
Architecture evolution

"*80 percent of the development effort in a big data project goes into data integration and only 20 percent goes toward data analysis.*" [Intel ETL White paper]

- Goal:
  - Scalable/Simplified/Mainstream
- Challenges:
  - Collect the raw data, process once and transform only when needed
    - More data (~ x10) to be archived and analyzed
    - Current PL/SQL jobs cannot make it
    - Hadoop/MapReduce seems appropriate, but it's a new processing paradigm and infrastructure

Lambda architecture

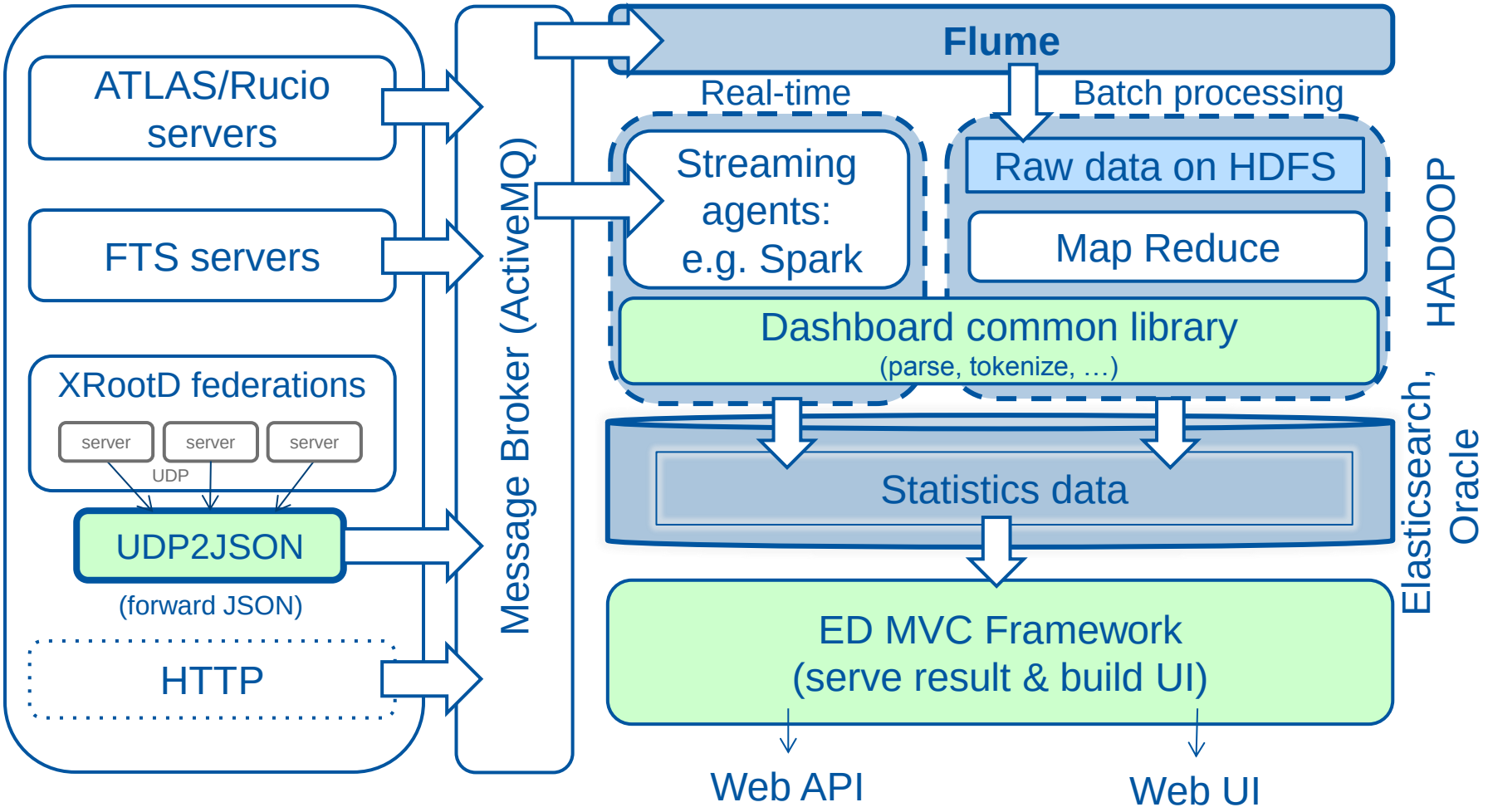
- Taking inspiration from the Lambda architecture
  - Implemented at Twitter
  - Intuitively, *one-technology-does-not-fit-all* idea
    - Batch for slow, reliable and stateless processing
    - Real-Time for fast, complex and incremental computation
    - Serve result from a dedicated serving layer



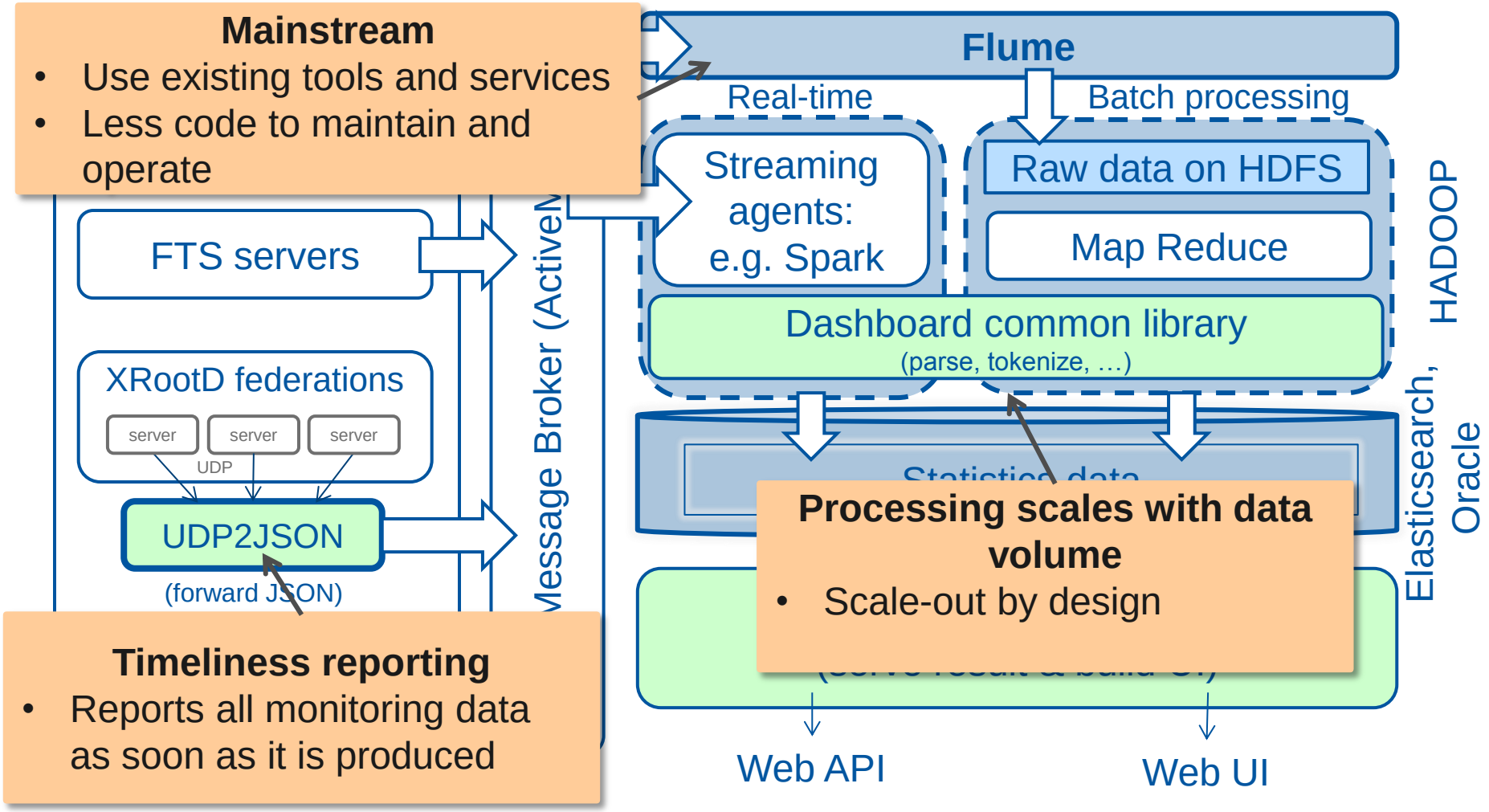
From the book:

Big Data
Principles and best practices of scalable realtime data system
By Nathan Marz and James Warren
ISBN: 9781617290343

New *lambda*-style architecture



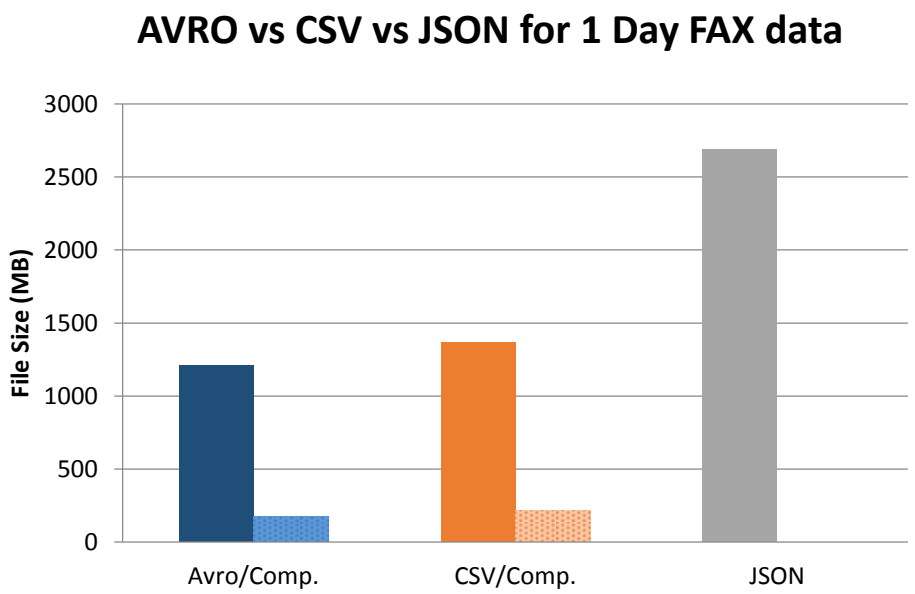
New *lambda*-style architecture



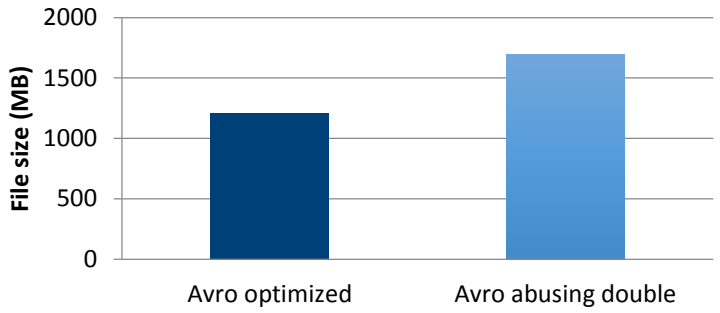
Now, make your choices!

How to store data on HDFS?

Do you need a schema? A serialization lib?



Is your schema *good* ?



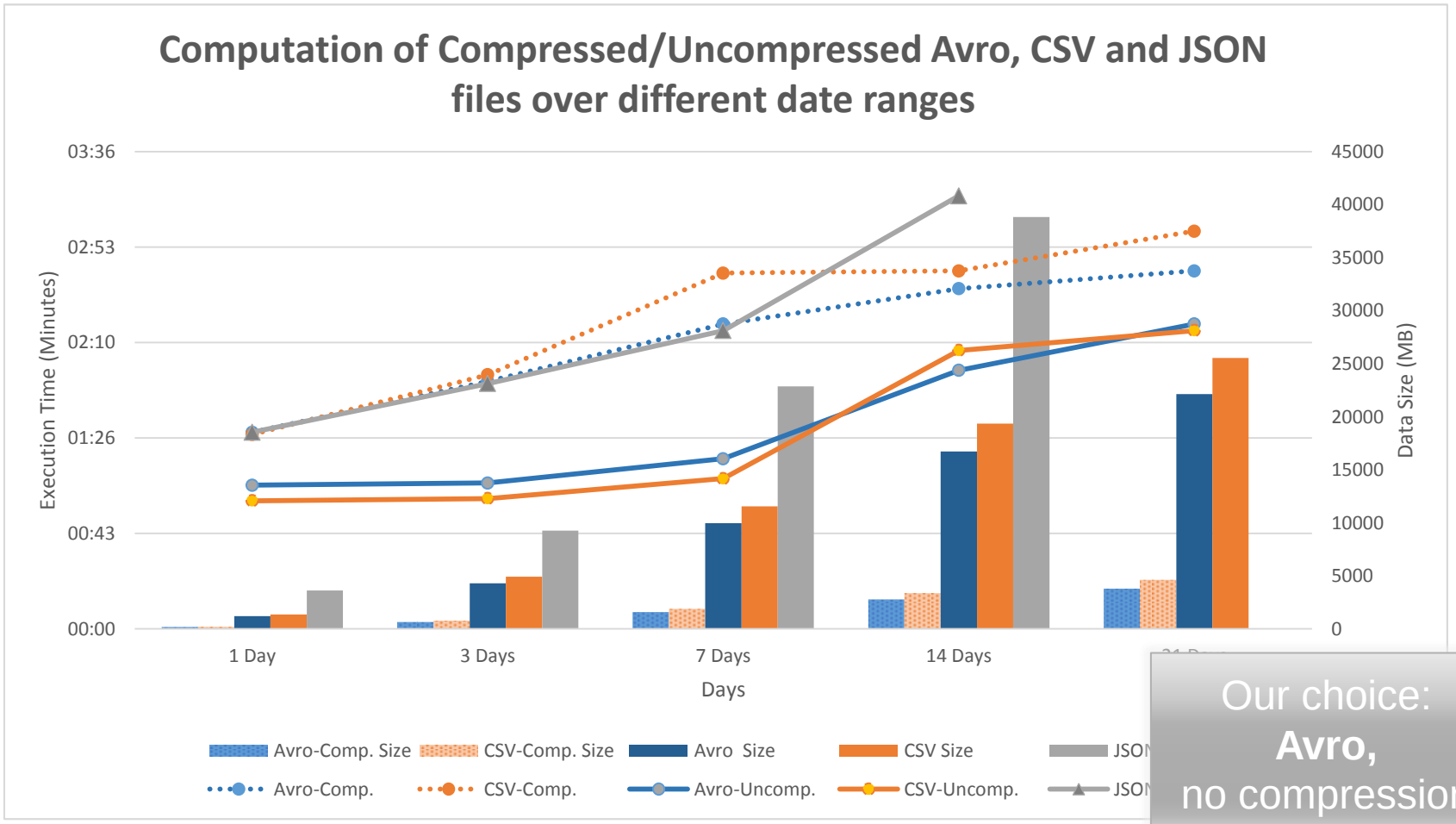
How to partition data?

```
|-- data
| |-- xrootd
| | |-- atlas
| | | |-- 2014
| | | | |--12
| | | | | |--01
| | | | | | | data.avro
```

Is compression a good idea?

Which algorithm?

Hadoop/MR* at work



Our choice:
Avro,
no compression

* CERN IT-DSS Analytix cluster (Hadoop/CDH 5.1), 8 nodes 32 cores/64GB, 7 nodes 4 cores/8GB

Real-Time layer: more choices!

- Streaming frameworks
  - Distributed (á la Hadoop)
  - Basic processing (filter, join, etc.)
  - Apache Storm, Apache Spark (Streaming), Apache Samza
- In-memory engines
  - Advanced processing (with DSLs like SQL or time interval algebra)
  - Esper, VoltDB
- We are currently investigating:
  - Apache Spark for streaming, potentially with Esper as operator for *complex* task

Conclusion

- Hadoop ecosystem fits well with WLCG monitoring
  - Scales with data volume
  - For Dashboard, imperative MR easier than SQL
  - Simple**r** architecture, less custom code and services
- The lambda idea enables "*live-views*" on WLCG
  - Keep all the data
  - Batch + streaming for fast and reliable processing
  - Ongoing investigation on Spark as ~ *uniform* platform
    - MR to Spark for batch trivial, done already (thanks to common lib)
- Plan: towards production
  - Migrate XRootD (FAX, AAA federations) and HTTP dashboards to new architecture by summer 2015
  - Focus then on FTS and DDM dashboard migration


CERN