

NXCALS - ARCHITECTURE AND CHALLENGES OF THE NEXT CERN ACCELERATOR LOGGING SERVICE

J. Wozniak, C. Roderick, CERN, Geneva, Switzerland

Ref. WEPHA163

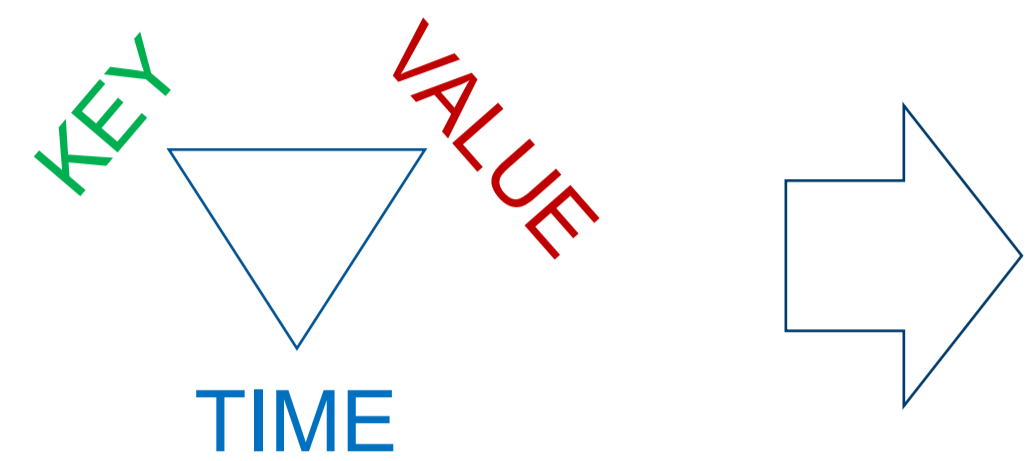
ABSTRACT

CERN Accelerator Logging Service (CALs) is in production since 2003 and stores data from accelerator infrastructure and beam observation devices. Initially expecting 1TB / year, the Oracle-based system has scaled to cope with 2.5TB / day coming from >2.3 million signals. It serves >1000 users making an average of 5 million extraction requests per day. (...) In 2016 the Next CERN Accelerator Logging Service (NXCALS) project was launched, with the aim of replacing CALs from Run 3 onwards using "Big Data" technologies. The NXCALS core is production-ready, based on open-source technologies such as Hadoop, HBase, Spark and Kafka. (...)



Fig 1. Data growth

DATA GROWTH
NO ANALYTICS API
DATED ARCHITECTURE
ORACLE NOT SCALABLE



TIMESERIES DATA PARTITIONING

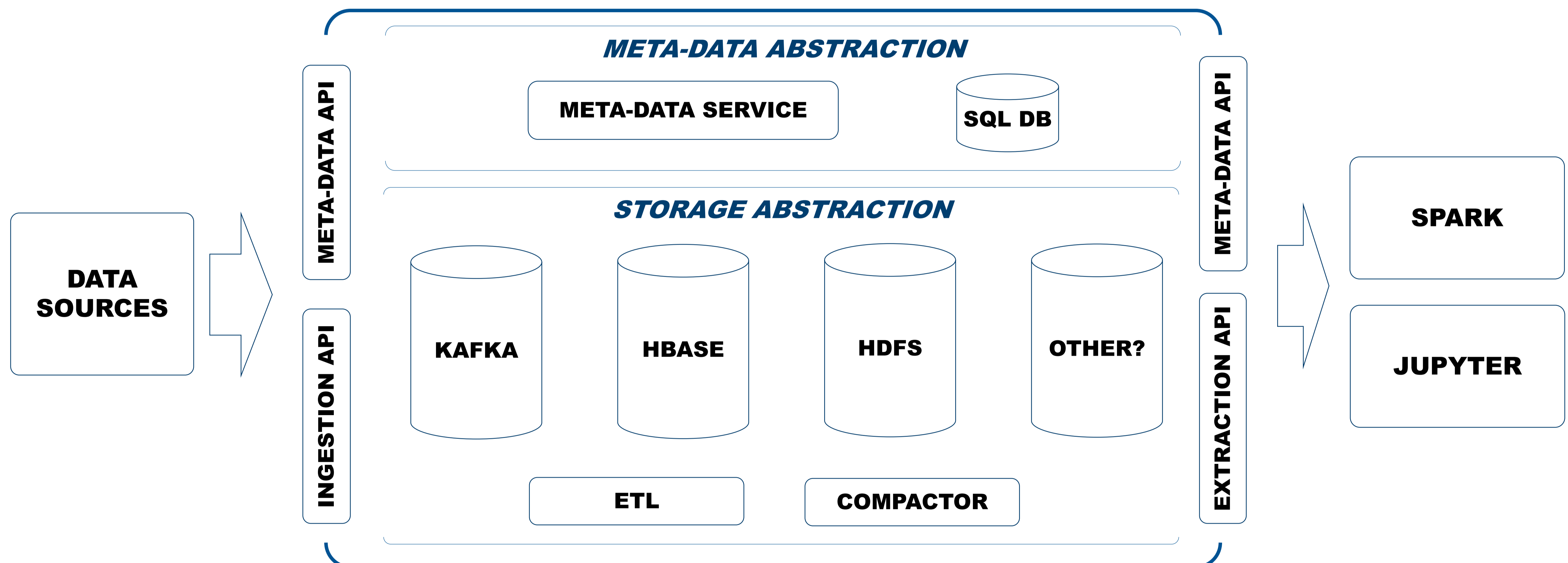
SYSTEM: (ENTITY_KEYS, PARTITION_KEYS, TIMESTAMP, FIELD₁...FIELD_N) = RECORD

HDFS: /// PROJECT / NXCALS / SYSTEM_ID / PARTITION_ID / SCHEMA_ID / DATE / DATA.PARQUET



DESIGN

ARCHITECTURE



DATA SENDING

```
Publisher<ImmutableData> publisher
= PublisherFactory
  .newInstance()
  .createPublisher("mySystem");

publisher.publish(ImmutableData.builder()
  .add("device", "myDevice")
  .add("property", "myProperty")
  .add("timestamp", Instant.now())
  .add("field1", 10L)
  .build());
```

5 PB HDD
8 TB RAM
20 NODES
960 CORES

DATA READING

```
SparkSession spark = ...

Dataset<Row> dataset = DataQuery.builder(spark)
  .byEntities()
  .system("mySystem")
  .startTime("2019-09-01 00:00:00")
  .endTime(Instant.now())
  .entity()
  .keyValueLike("device", "myDev%")
  .keyValue("property", "myProperty")
  .buildDataset();

dataset.count();
```

USAGE

DATA MIGRATION

100 DATA NODES
2.1e6 REC/SEC
3.2e13 REC TOTAL



Fig 2. Migration rec/sec rate

SCALABLE DESIGN
PRODUCTION READY
NOT CERN SPECIFIC

