Intel Big Data Analytics

openlab Technical Workshop

Evangelos Motesnitsalis

23 January 2019

openlab Big Data Analytics

in collaboration with Intel

SERVICE IT-DB

Provides infrastructure, knowledge, consultancy, and integration with the rest of the IT Services

Ensures that Industry, CERN IT, and the Experiments are effectively connected

Provides resources and consultancy on big data technologies and optimizations

Provides the relevant use cases and physics analysis

openlab Big Data Analytics

in collaboration with Intel

The project aims at helping and optimizing
the Data Analytics Solutions at CERN in the areas of:

- Data Integration
- Data Ingestion and Transformation,
- Performance, Scalability, and Benchmarking
- Resource Management
- Data Visualization
- Hardware Utilization

Motivation and Vision

Why Big Data Analytics for High Energy Physics?

Investigate new ways to analyse physics data in preparation for the HL-LHC

Adopt new technologies widely used in the industry
- Open the HEP field to a larger community of data scientists
- Bring together engineers from industry and domain experts from academia

Use modern APIs, development environments and platforms (notebooks etc.)

Allow further development with Streaming and Machine Learning workloads

HEP Data Processing

Physics Analysis is typically done with the ROOT Framework which uses physics data that are saved in ROOT format files. At CERN these files are stored within the EOS Storage Service.

EOS Service

A disk-based, low-latency storage service with a highly-scalable hierarchical namespace, which enables data access through the XRootD protocol.

ROOT Data Analysis Framework

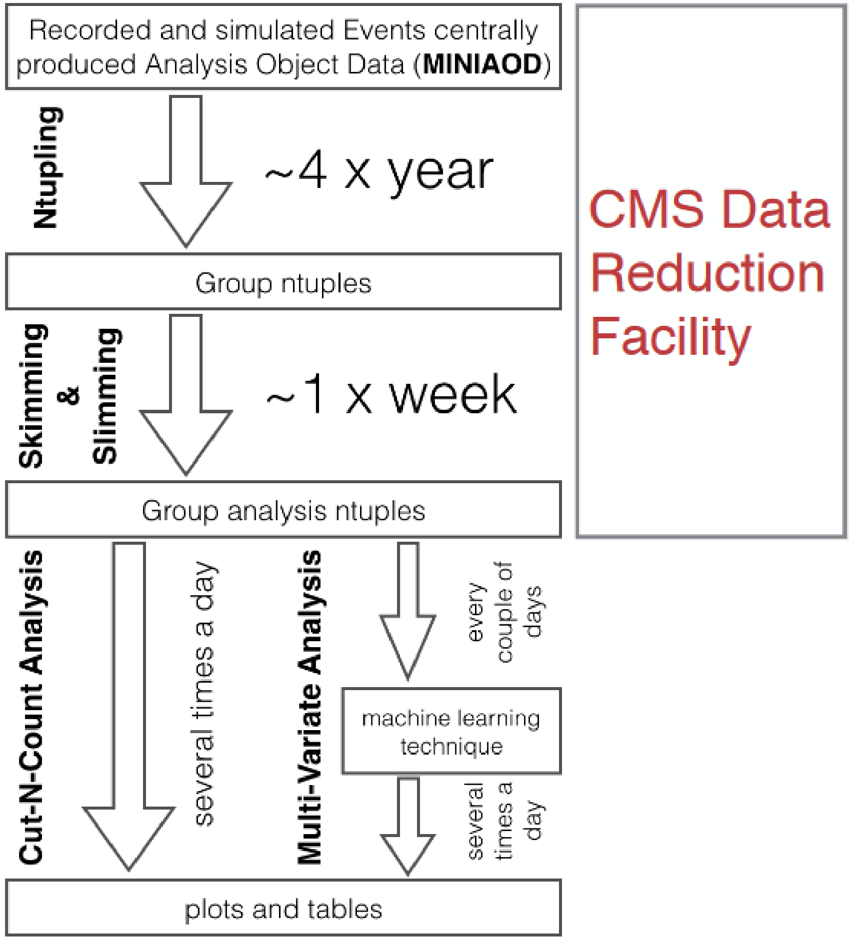
A modular scientific software framework which provides all the functionalities needed to deal with big data processing, statistical analysis, visualization and file storage.

CMS Data Reduction and Analysis Facility

Performing Physics Analysis and Data Reduction with Apache Spark

- Investigate new ways to analyse physics data and improve resource utilization and time-to-physics
- Data Reduction refers to event selection and feature preparation based on potentially complicated queries
- The IT Hadoop and Spark Service has the capacity to run this type of jobs
- Main goal was to be able to reduce **1** PB of data in 5 hours or less
- We now have fully functioning Analysis and Reduction examples tested over CMS Open Data
- We performed extensive scaling and performance optimizations

CMS Data Reduction and Analysis Facility



- It offers an alternative for 'ad-hoc' data reduction for each research group
- This type of facility could be a big shift for High Energy Physics
- Bridge the gap between High Energy Physics and Big Data communities

Apache Spark

Apache Spark is an open source cluster computing framework

Runs on:
- Hadoop
- HPC
- Cloud

APIs in Python, Scala, Java, R

Compatible with multiple cluster managers:
- Apache YARN
- Apache Mesos
- Kubernetes
- Standalone

Multiple File Formats and Filesystem Compatibility

Consists of multiple components:
- Spark SQL
- Spark Mlib
- Spark Graph
- Spark Structured Streaming

Milestones and Achievements

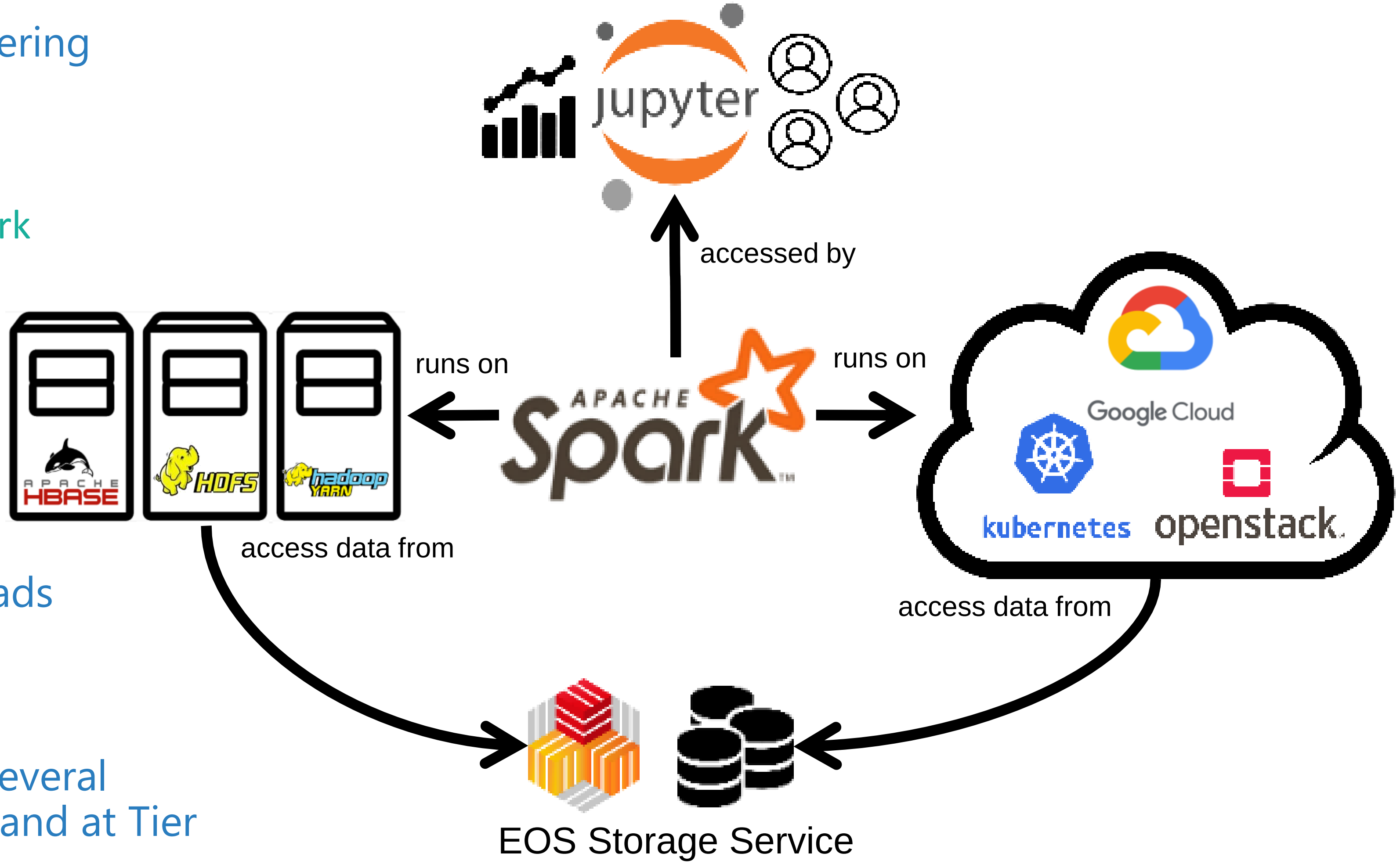
We solved two important data engineering challenges:

1. Read files in ROOT Format using Spark

2. Access files stored in EOS directly from Hadoop/Spark

This enabled us to produce, scale up, and optimize Physics Analysis Workloads with input up to 1 PB.

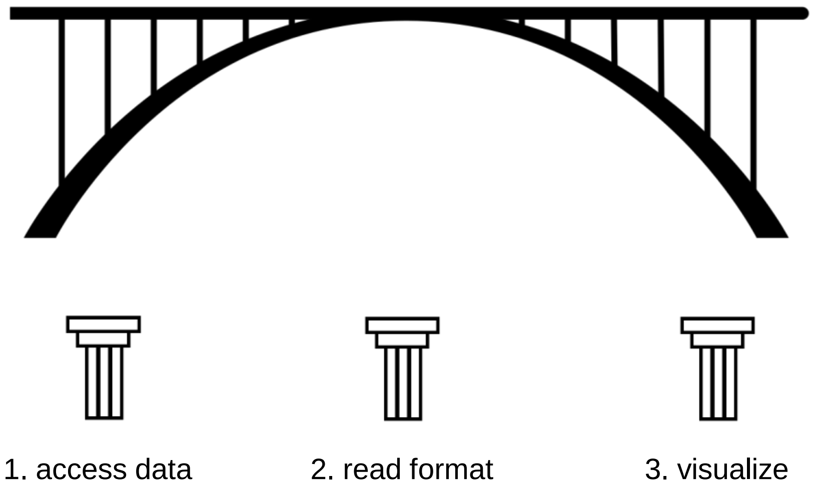
The infrastructure is actively used by several physics analysis groups both at CERN and at Tier 1s.



Bridging the Gap

Physics Analysis is typically done with the ROOT Framework which uses physics data that are saved in ROOT format files. At CERN these files are stored within the EOS Storage Service.

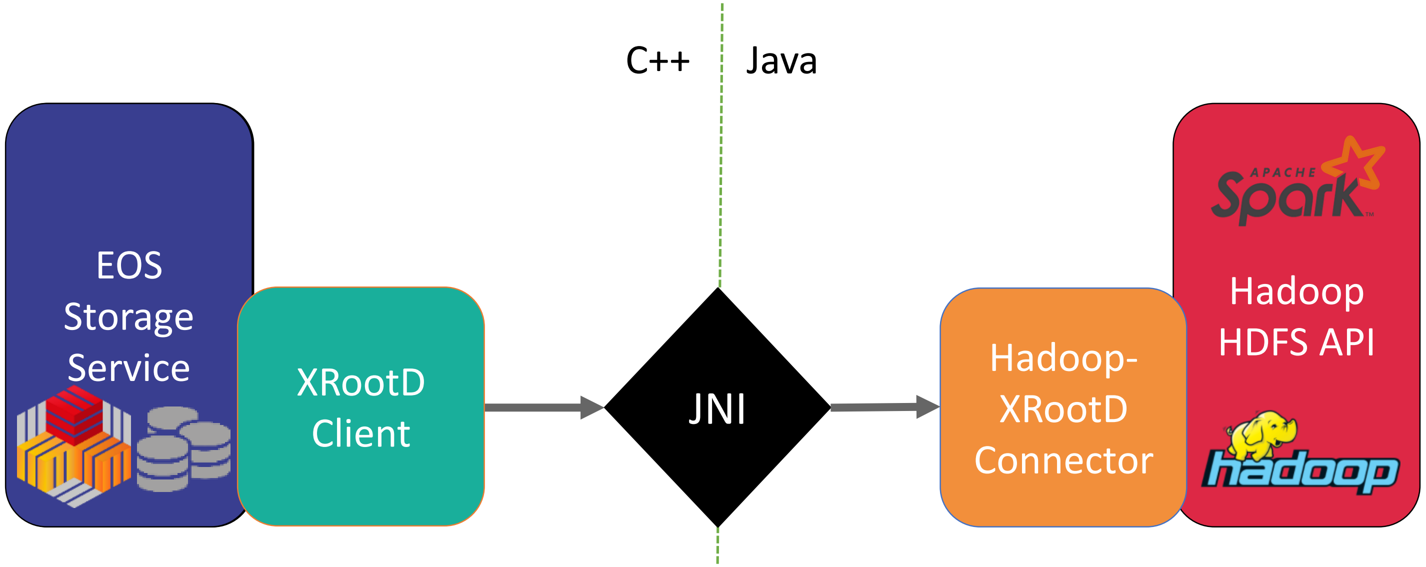
EOS Storage Service



Hadoop – XRootD Connector

Connecting XRootD-based Storage Systems with Hadoop and Spark

- A Java library that connects to the XRootD client via JNI
- Reads files from the EOS Storage Service directly
- Makes all Physics Data available for processing with Spark
- Supports Kerberos and GRID Certificate Authentication



Open Source: https://github.com/cerndb/hadoop-xrootd

Spark - Root

A Scala library which implements DataSource for Apache Spark

Spark can read ROOT TTrees and infer their schema

Root files are imported to Spark Dataframes/Datasets/RDDs

Developed by DIANA-HEP in collaboration with CERN openlab

Open Source: https://github.com/diana-hep/spark-root/

SWAN Service and Spark Integration

Hosted Jupyter Notebooks for Data Analysis

- Web-based interactive analysis using PySpark in the cloud
- Collaboration between EP-SFT, IT-ST, and IT-DB
- No need to install software
- Combines code, equations, text and visualisations
- Direct access to the EOS and HDFS
- Fully Integrated with IT Spark and Hadoop Clusters

https://swan.web.cern.ch/

Scalability Tests

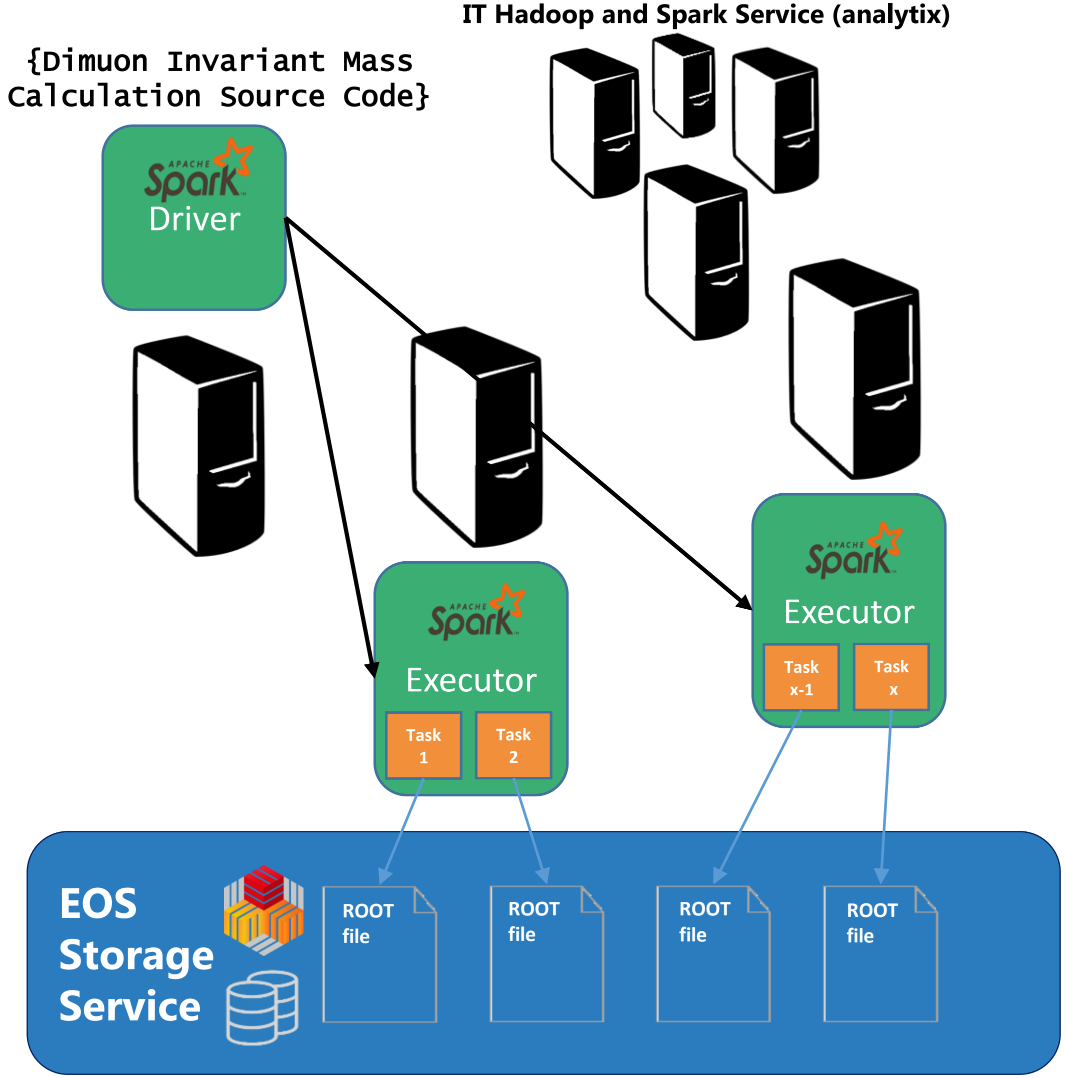
The data processing job of this project was developed in Scala by CMS members.

Its code:

- Performs event selection (i.e. Data Reduction)
- Uses the filtered events to compute the dimuon invariant mass

On a single thread/core and one single file as input, the workload reads one branch and calculates the dimuon invariant mass in approximately 10 mins for a 4GB file

Test Workload Architecture and File-Task Mapping



Open Source: https://github.com/olivito/spark-root-applications

Scalability Tests

Technologies used:

- Intel CoFluent Cluster Simulation Technology
- Apache Spark
- Hadoop YARN
- Kubernetes and Openstack

Issues that we had to tackle:

- Network bottleneck
- "readAhead" buffer configuration
- Running tests on a shared cluster can lead to resources denial
- Performance impact on IT production services

Services/Tools Used:

- EOS Public, EOS UAT
- Hadoop-XRootD Connector
- spark-root
- sparkMeasure
- Spark on Kubernetes Service

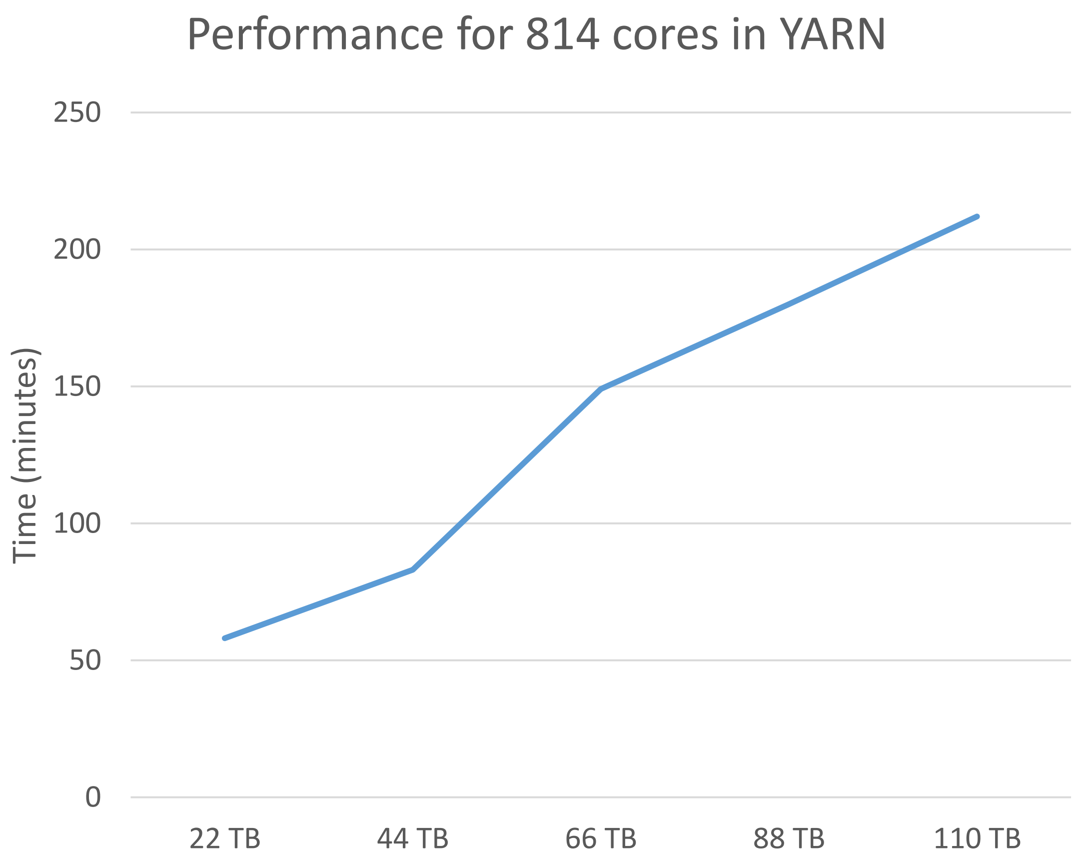
We collaborated with:

- IT-ST and the EOS service for a dedicated EOS instance (UAT)
- IT-DB for a dedicated queue in YARN with prereserved resources
- IT-CM and the OpenStack service for a dedicated Kubernetes cluster on OpenStack

Scalability Tests – Phase I

Performance and Scalability for different input size with 32 MB “readAhead” buffer, 814 logical cores, and 2 logical cores per Spark executor

Input Data	Time for EOS Public
22 TB	58 mins
44 TB	83 mins
66 TB	149 mins
88 TB	180 mins
110 TB	212 mins

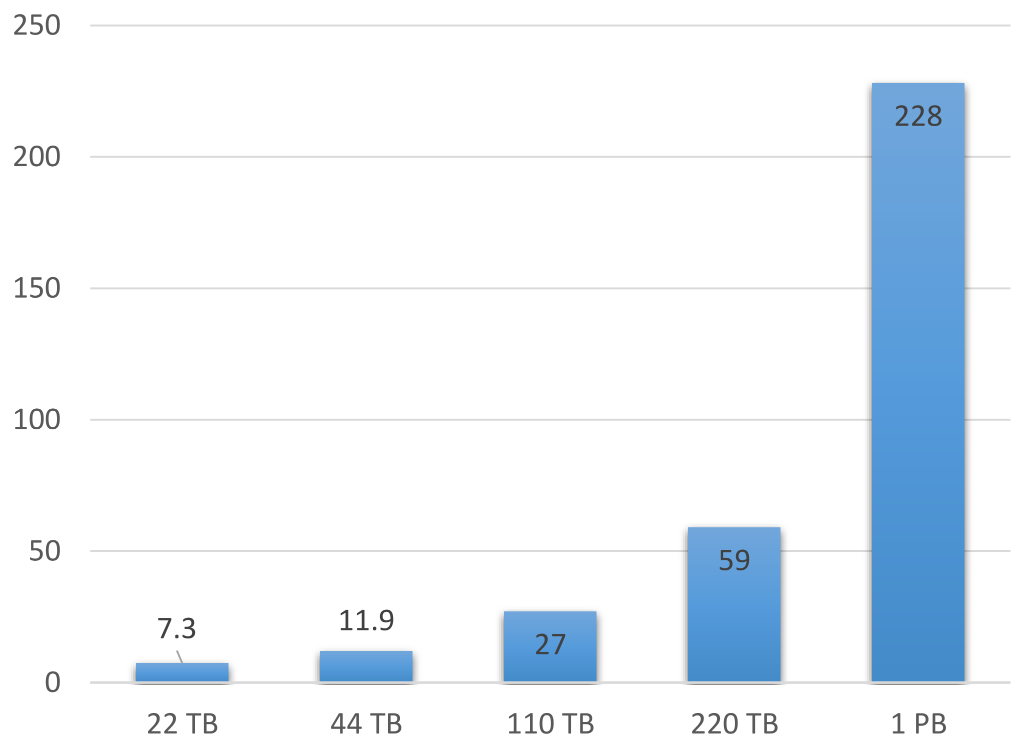


Configuration: 407 executors, 2 cores per executor, 7 GB per executor

Scalability Tests – Phase II

Performance and Scalability of the tests for different input size in minutes with 64 KB of "readAhead" buffer, 800 logical cores, and 8 logical cores per Spark executor

Input Data	Time for EOS Public
22 TB	7.3 mins
44 TB	11.9 mins
110 TB	27 mins (±2)
220 TB	59 mins (±5)
1 PB	228 mins (±10) (~3.8 hours)

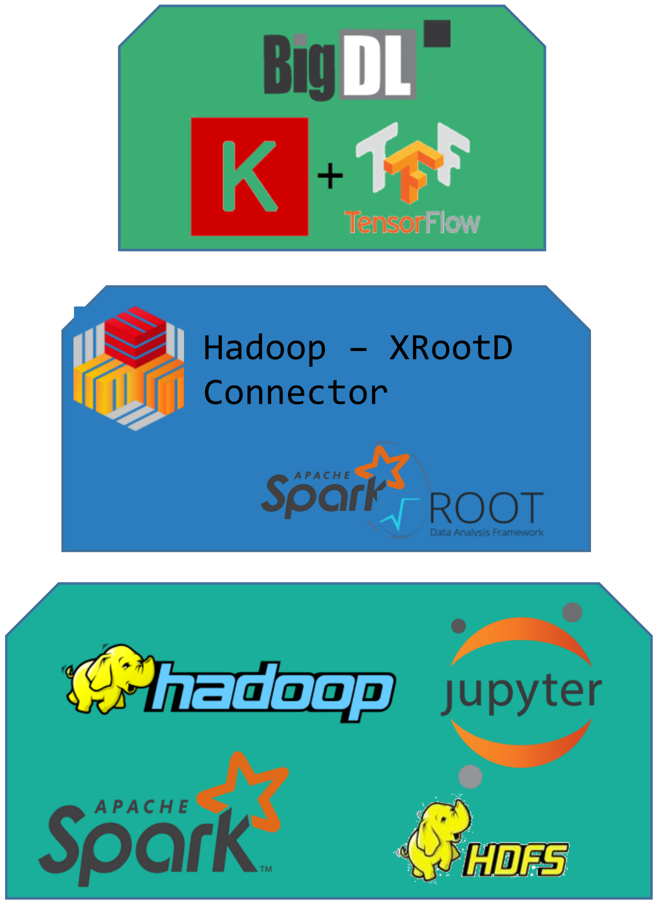


Configuration: 100 executors, 8 cores per executor, 7 GB per executor

CERN openlab

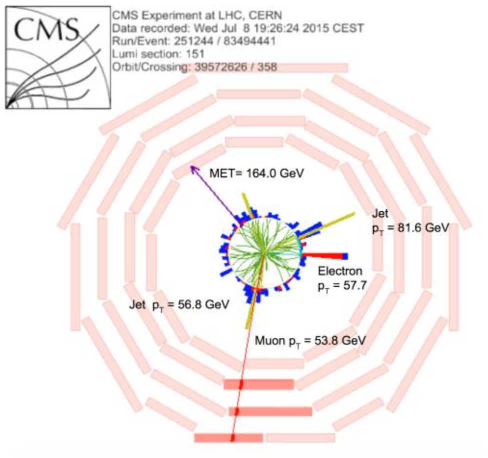
More details in the poster "Physics Data Analysis and Data Reduction at scale with Apache Spark"

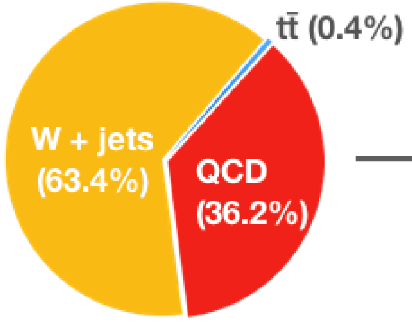
From Data Engineering to Machine Learning



Machine Learning Workloads

Topology classification with deep learning to improve real time event selection at the LHC
[https://arxiv.org/abs/1807.00083]





Topology Classifier → W+j / QCD / tt

Particle-sequence classifier
Recurrent Neural Network taking as input a list of particles

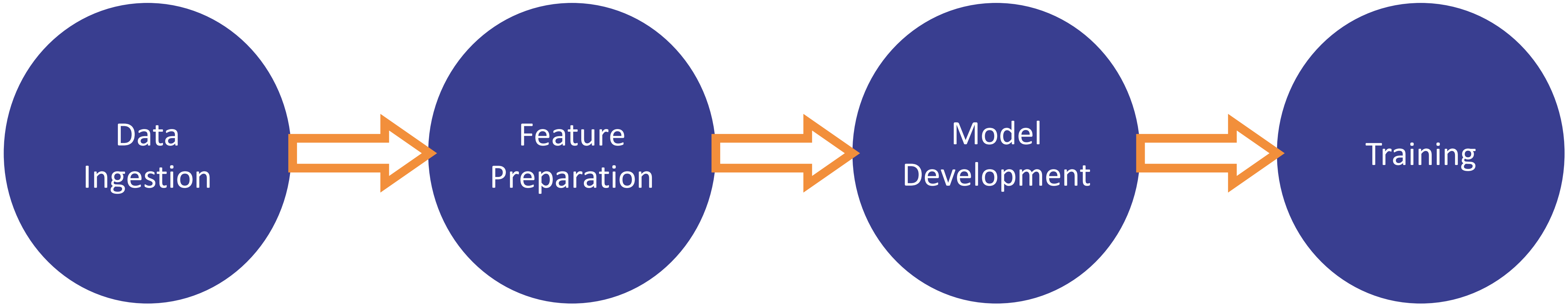
High level feature classifier
Fully connected neural network taking as input a list of features computed from low level features

Inclusive classifier
Particle-sequence classifier + information coming from the HLF

Machine Learning Workloads

The goals of this work are:

- Produce an example of a ML pipeline using Spark + EOS and ROOT format integration
- Test the performances of Spark at each stage for this use case



- Read Root Files from EOS
- Produce HLF and LLF datasets

- Produce the input for each classifier

- Find the best model using Grid/Random search

- Train the best model on the entire dataset

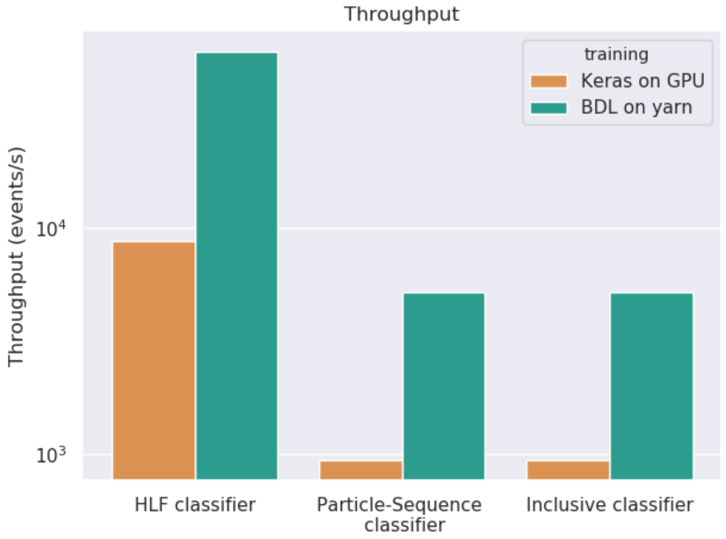
CERN openlab

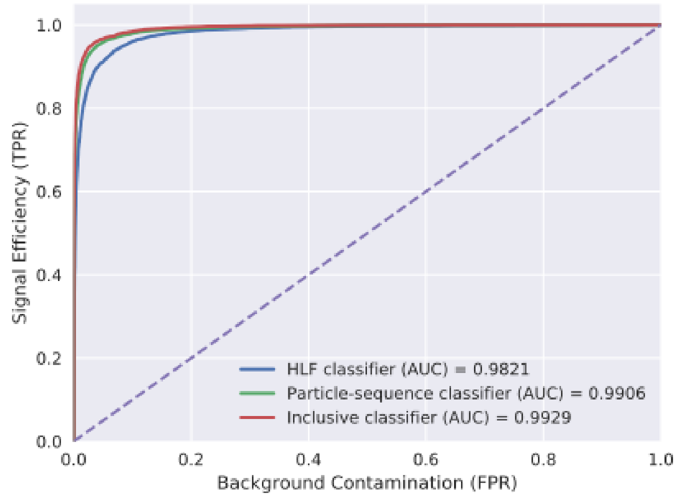
Machine Learning Workloads

Created an End-to-End scalable machine learning pipeline using Apache Spark and industry standard tools

BigDL + Spark on CPU performs and scales well for recurrent NN and deep NN

Compatible with results presented in the paper





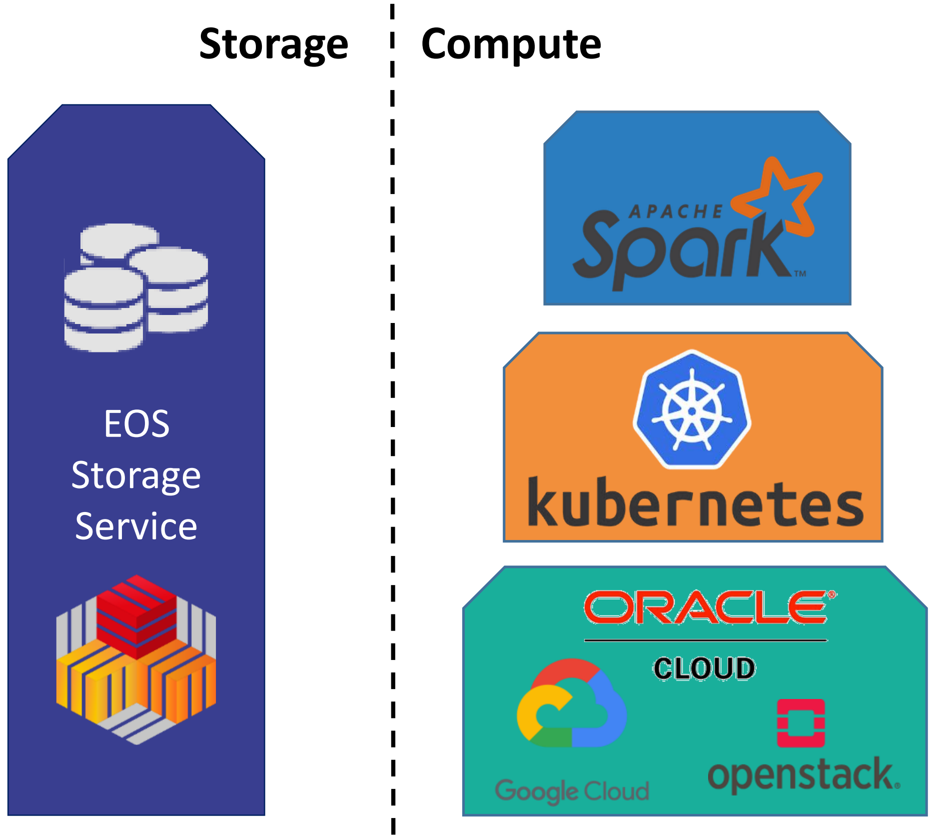
CERN openlab

More details in the poster "Machine Learning Pipelines with Apache Spark and Intel BigDL"

Apache Spark on Cloud

Work in Progress

- Cloud Architecture for data analysis means separating storage and computing
- Allow users to access a larger environment of libraries developed by different communities
- Provide an abstraction to run Spark and popular Machine Learning frameworks with Kubernetes independently from the underlying resources



More details in the poster "Physics Data Processing and Machine Learning on the Cloud"

Future Steps

- Repeat the workload tests on top of virtualized/containerized infrastructure with Kubernetes in bigger infrastructure and public clouds
- Extend the workloads to different and more complex use cases of Physics Analysis
- Proceed with more Machine Learning and Online Data Processing (Streaming) use cases
- Extend the features of the "Hadoop-XRootD Connector" library (i.e. write to EOS, better packaging, monitoring)

Conclusions

Can we reduce 1 PB in 5 hours?

- Yes, we even dropped to 4 hours in our latest tests.

Through this project we achieved:

- Efficient & fast processing of physics data
- Connecting Libraries between Big Data Technologies and HEP Tools
- Adoption of Big Data Technologies by CMS physics groups (e.g. University of Padova, Fermilab)

Future Plans:

- Focus on Machine Learning and Streaming workloads
- Consolidate and extend the existing libraries

Acknowledgements

CERN openlab

Colleagues at the CERN Hadoop, Spark and streaming services.

Intel

CMS members of the Big Data Reduction Facility, DIANA/HEP, Fermilab as well as the authors of “Topology classification with deep learning to improve real-time event selection at the LHC” [https://arxiv.org/abs/1807.00083]

Questions?

emotes@cern.ch