CERN openlab

Physics Data Analytics and Data Reduction with Apache Spark

Evangelos Motesnitsalis

11 October 2017

Contents

Goal

Goal

“to perform Physics Analysis and Data Reduction
with Big Data Technologies
over data acquired from the CMS Experiment”

Motivation and Vision

Why Data Analytics and Reduction with Spark?

Motivation and Vision

Why Data Analytics and Reduction with Spark?

- Investigate new ways to analyse physics data and improve resource utilization and time-to-physics

Motivation and Vision

Why Data Analytics and Reduction with Spark?

- Investigate new ways to analyse physics data and improve resource utilization and time-to-physics

- Adopt new technologies widely used in the industry
  - Open the HEP field to a larger community of data scientists
  - Improve chance of researchers on the job market outside academia

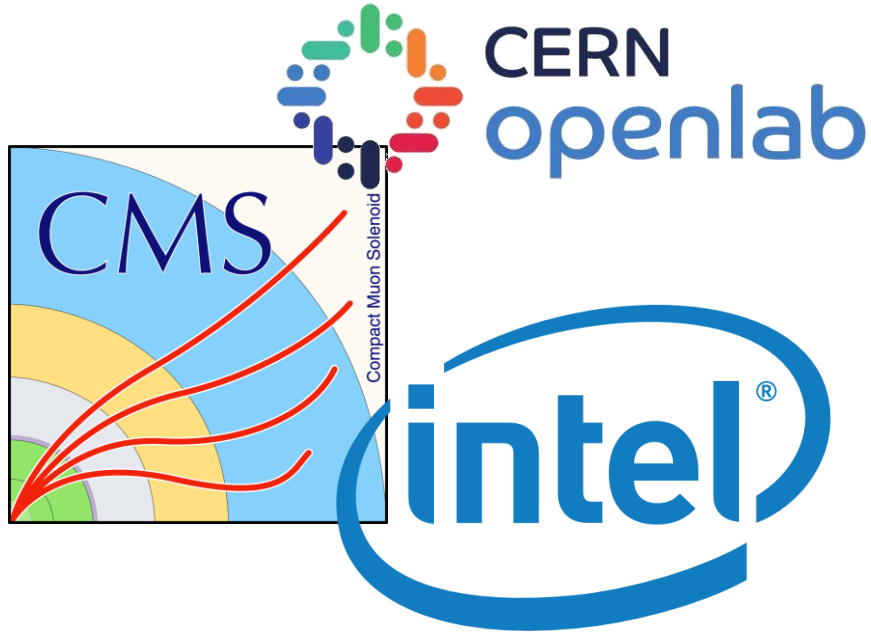
Collaboration Members

Who is participating?

Collaboration Members

Who is participating?

- CERN IT Department (openlab and IT-DB)
- The CMS Experiment
- Intel



Standard HEP Analysis Procedures

How is Physics Analysis done currently?

Standard HEP Analysis Procedures

How is Physics Analysis done currently?

Until today, the vast majority of high energy physics analysis is done with the ROOT Framework which uses physics data that are stored in ROOT format files, within the EOS Storage Service of CERN.

Standard HEP Analysis Procedures

How is Physics Analysis done currently?

Until today, the vast majority of high energy physics analysis is done with the ROOT Framework which uses physics data that are stored in ROOT format files, within the EOS Storage Service of CERN.

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. It is mainly written in C++ but also integrated with Python and R.

Standard HEP Analysis Procedures

How is Physics Analysis done currently?

Until today, the vast majority of high energy physics analysis is done with the ROOT Framework which uses physics data that are stored in ROOT format files, within the EOS Storage Service of CERN.

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. It is mainly written in C++ but also integrated with Python and R.

EOS Service

A disk-based, low-latency storage service with a highly-scalable hierarchical namespace, which enables data access through the XROOT protocol. It provides storage for both physics and user use cases via different service instances such as EOSPUBLIC, EOSATLAS, EOSCMS.

Progress

What has been done to date?

Progress

What has been done to date?

Two Main Challenges have been solved:

1. Read files in ROOT Format using Spark

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

Progress

What has been done to date?

Two Main Challenges have been solved:

1. Read files in ROOT Format using Spark

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

We now have fully functioning Analysis & Reduction examples tested over CMS Open Data (1 TB)

1. Read Files in ROOT Format directly from Spark

DIANA-HEP developed “spark-root”

A Scala library which implements DataSource for Apache Spark

Spark can read ROOT TTrees and infer their schema

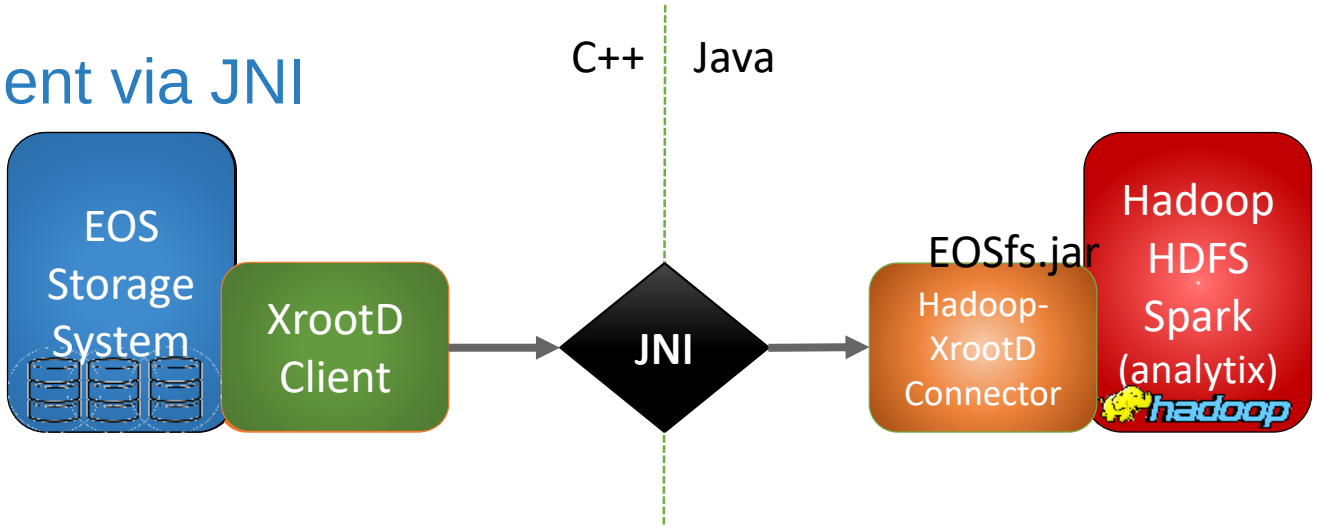
Root files are imported to Spark Dataframes/Datasets/RDDs

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

We developed “Hadoop-XrootD Connector”

A library that connects to the XrootD client via JNI

It reads files on EOS directly
(no need for import/export)



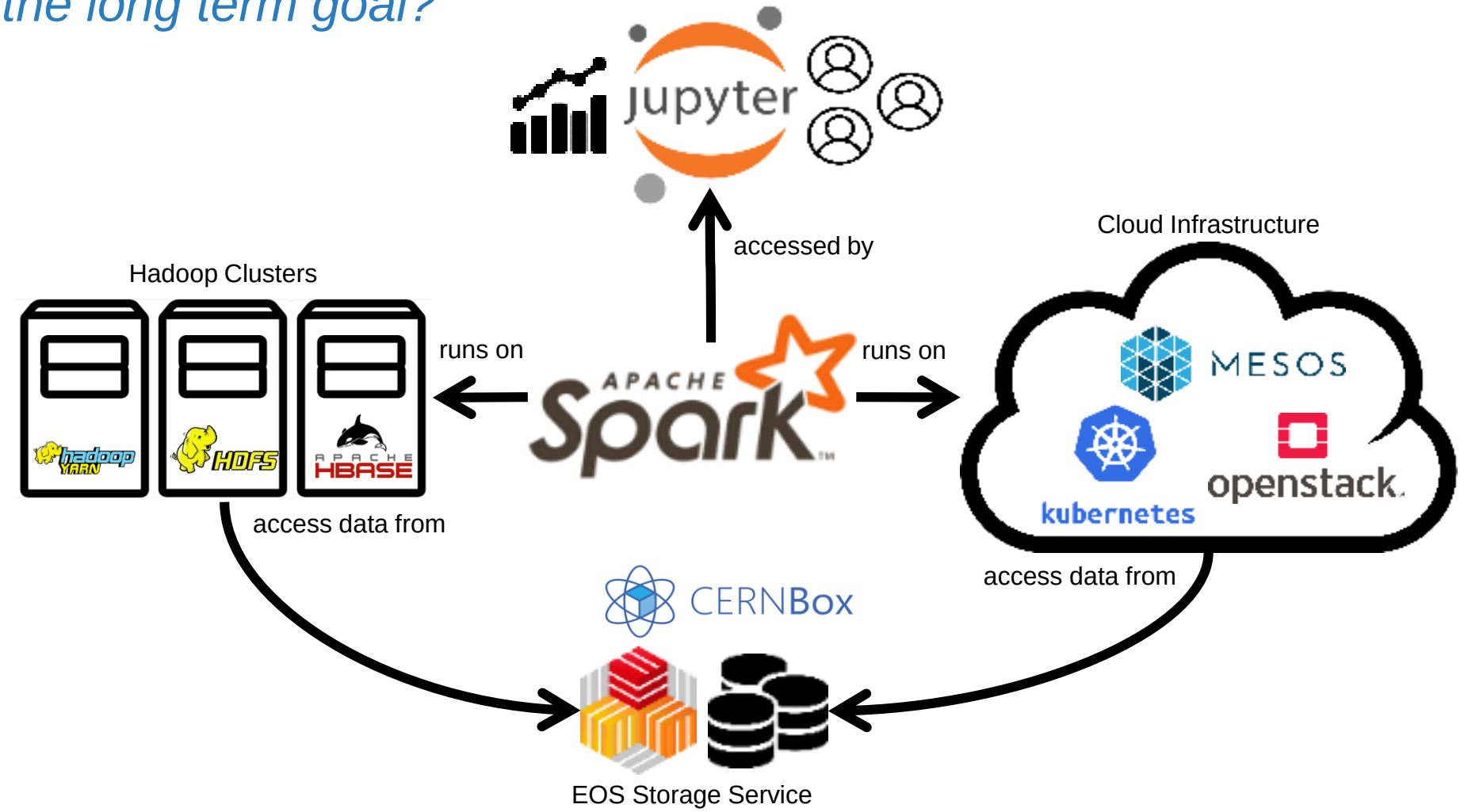
Currently working with 30% the speed of direct HDFS access for a READ operation

Planned Architecture Overview

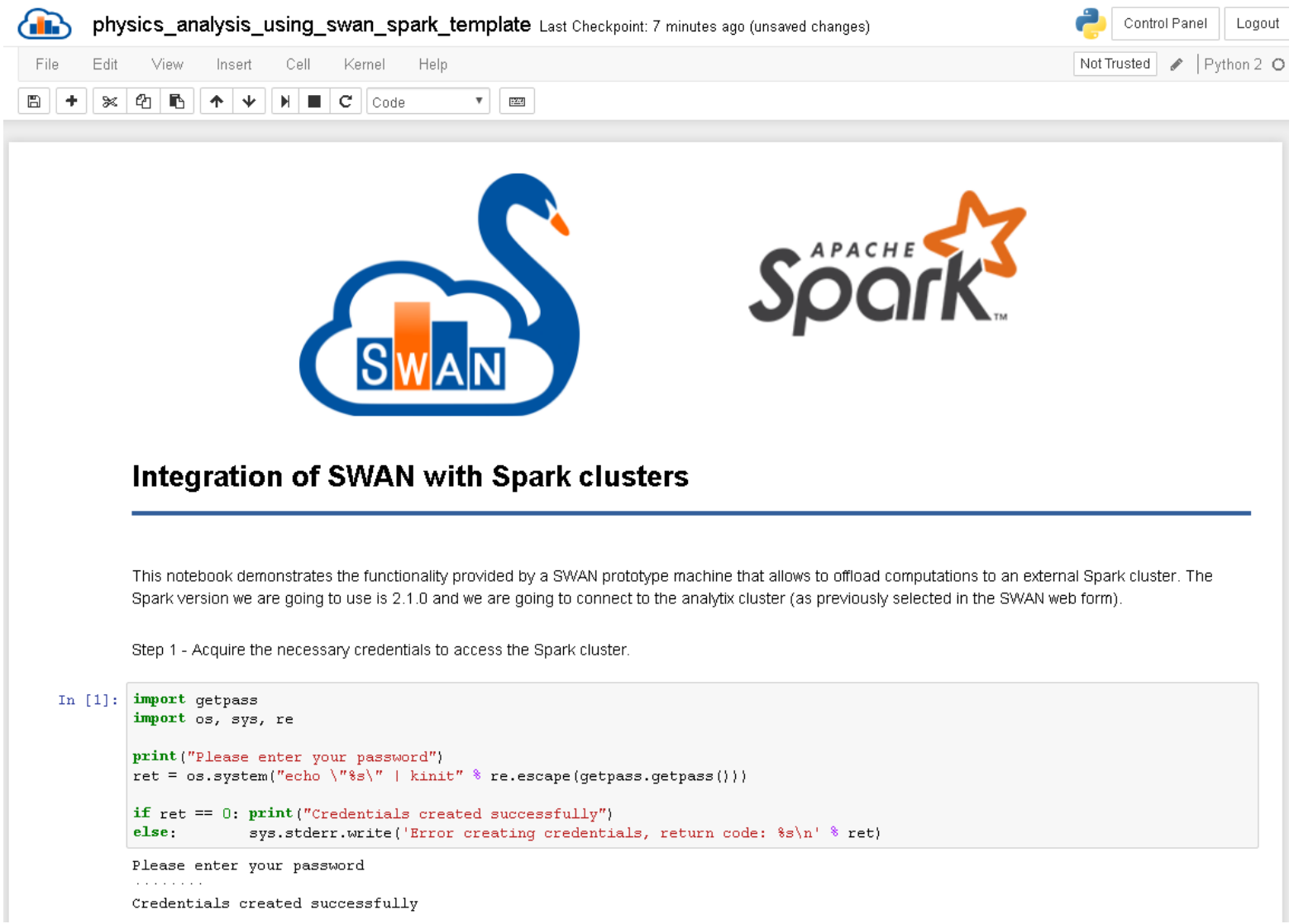
What is the long term goal?

Planned Architecture Overview

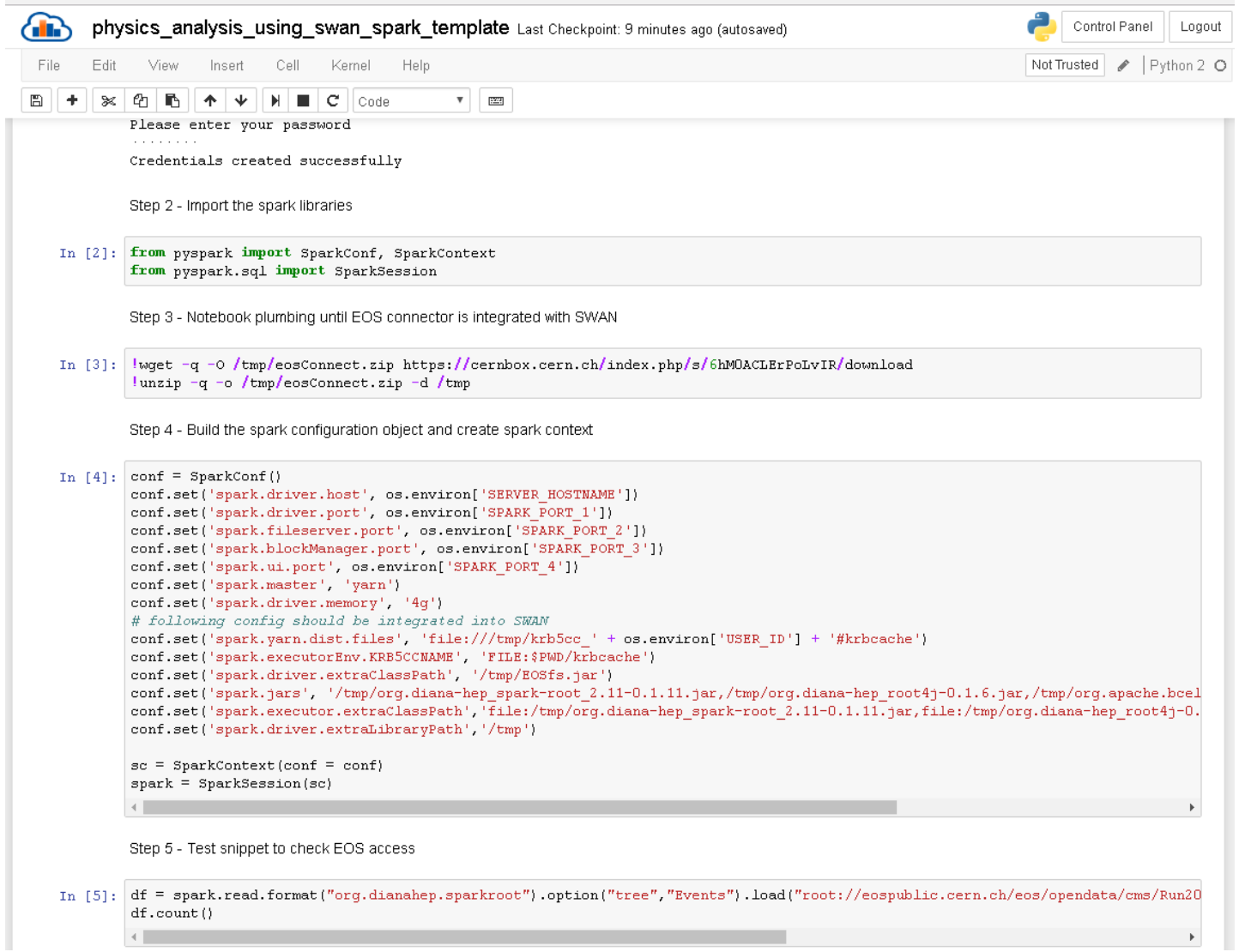
What is the long term goal?



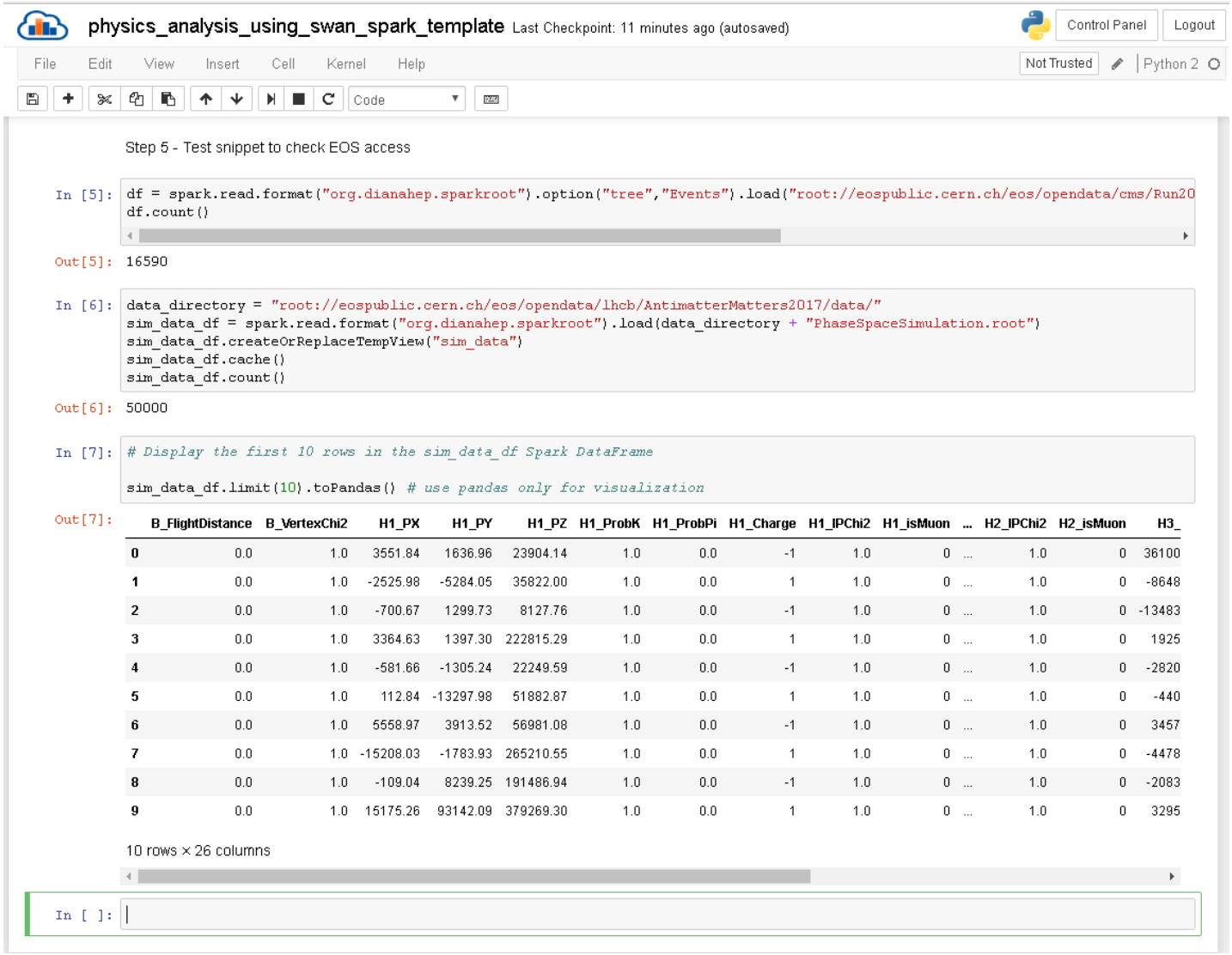
Lightning Demo



Lightning Demo



Lightning Demo



Future steps

What is next?

Future steps

What is next?

Hadoop-XrootD Connector:

- Add GRID Authentication alongside with Kerberos
- Introduce optimizations and tuning to decrease the performance gap between EOS and HDFS

Future steps

What is next?

Hadoop-XrootD Connector:

- Add GRID Authentication alongside with Kerberos
- Introduce optimizations and tuning to decrease the performance gap between EOS and HDFS

spark-root:

- Implement the ability to write to ROOT files
- Add IO statistics support

Future steps

What is next?

Hadoop-XrootD Connector:

- Add GRID Authentication alongside with Kerberos
- Introduce optimizations and tuning to decrease the performance gap between EOS and HDFS

spark-root:

- Implement the ability to write to ROOT files
- Add IO statistics support

Investigate scaling behavior for larger input (goal is 1 PB)

Future steps

What is next?

Hadoop-XrootD Connector:

- Add GRID Authentication alongside with Kerberos
- Introduce optimizations and tuning to decrease the performance gap between EOS and HDFS

spark-root:

- Implement the ability to write to ROOT files
- Add IO statistics support

Investigate scaling behavior for larger input (goal is 1 PB)

Investigate the possibility to use Spark over Openstack

Summary and Outlook

Summary and Outlook

We have solved two important challenges:

- Hadoop-XrootD Connector can directly access files from the EOS Service
- Spark-root can read ROOT files and infer their schema into Spark

Summary and Outlook

We have solved two important challenges:

- Hadoop-XrootD Connector can directly access files from the EOS Service
- Spark-root can read ROOT files and infer their schema into Spark

Our long term goal is to provide a unified Spark service:

- accessible via Notebook Software
- over Hadoop Clusters or over Openstack with Kubernetes or Apache Mesos

Summary and Outlook

We have solved two important challenges:

- Hadoop-XrootD Connector can directly access files from the EOS Service
- Spark-root can read ROOT files and infer their schema into Spark

Our long term goal is to provide a unified Spark service:

- accessible via Notebook Software
- over Hadoop Clusters or over Openstack with Kubernetes or Apache Mesos

This will simplify the existing physics data analysis procedures by:

- improving resource utilization
- providing user-friendly environments
- reducing the overall “time-to-physics”

Questions?

emotes@cern.ch