SWAN: Powering CERN's Data Analysis and Machine Learning Use cases

Riccardo Castellotti

On behalf of the SWAN team

https://swan.cern.ch

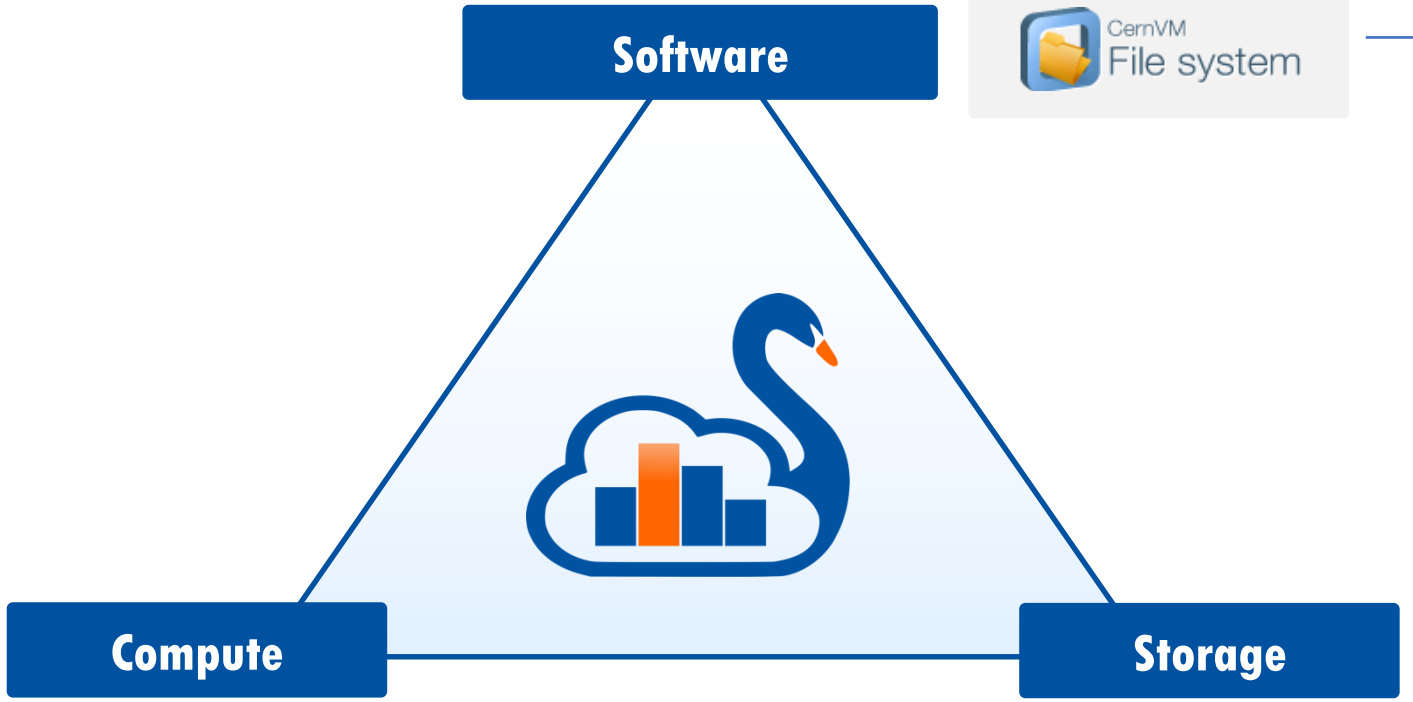
Oct 22nd, 2020

4th IML Machine Learning Workshop

SWAN in a Nutshell

- Data analysis with a web browser
  - No local installation needed
  - Based on Jupyter Notebooks
  - Calculations, input data and results “in the Cloud”
- Support for multiple analysis ecosystems and languages
  - Python, ROOT C++, R and Octave
- Easy sharing of scientific results: plots, data, code
- Already in use since 2016
  - 2000+ unique users in Physics, Accelerators and IT

Integrating services



CernVM File system

TensorFlow

NVIDIA CUDA

PYTORCH

APACHE Spark

docker

jupyter

APACHE hadoop

Isolation| local compute

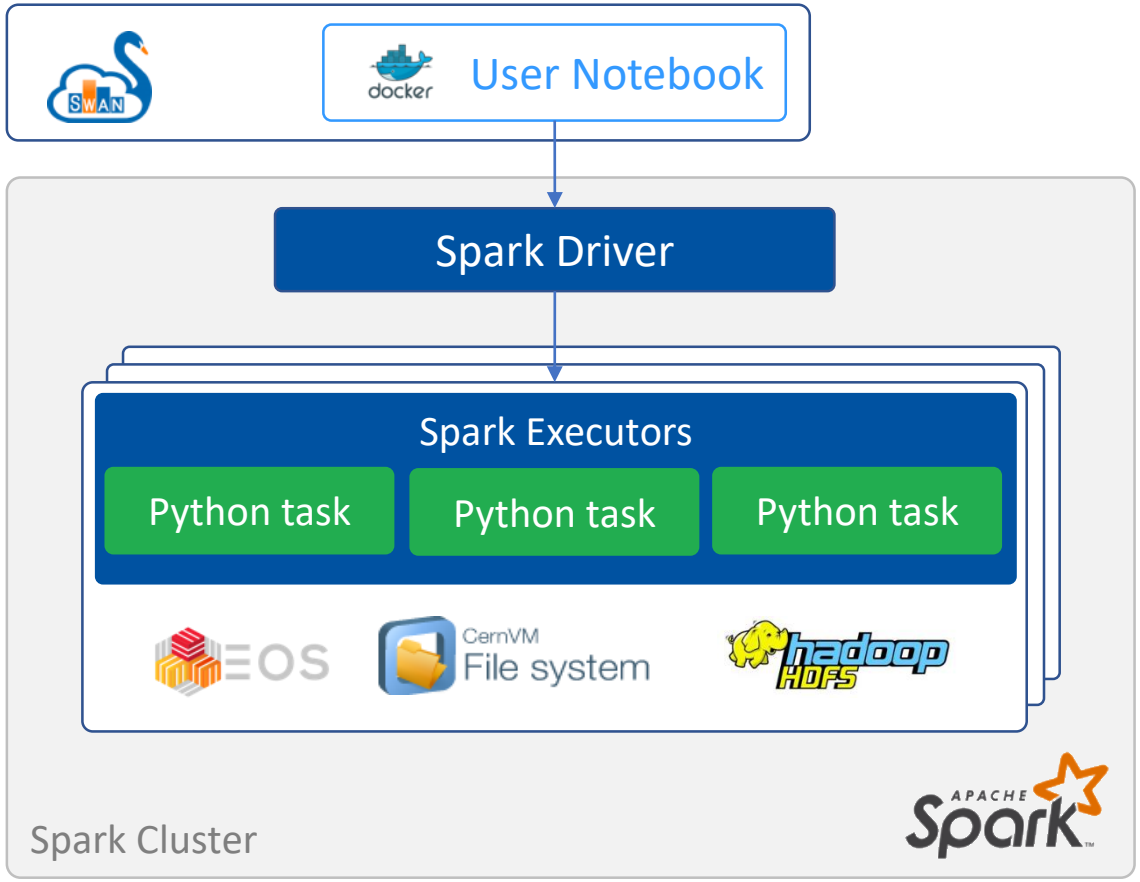
kubernetes

CERNBox

EOS

Integration with Apache Spark

- Connection to CERN Spark Clusters
  - Spark: general purpose distributed computing framework
- Same environment across platforms (local/remote)
  - Software - CVMFS
- Graphical Jupyter extensions developed
  - Spark Connector
  - Spark Monitor
- Spark Clusters
  - NXCals: – Dedicated cluster for accelerator logging
  - Analytix: – General purpose YARN cluster
    - For HDFS access
  - Cloud Containers: – General purpose Kubernetes cluster
    - For non-local storage (EOS)



ML pipelines with Spark + TensorFlow

Input: labeled data and DL models

Feature engineering at scale

Hyperparameter optimization (Random/Grid search)

Distributed model training

Output: particle selector model

[Machine Learning Pipelines with Modern Big Data Tools for High Energy Physics](#) M. Migliorini, R. Castellotti, L. Canali, E. Zanetti *Comput Softw Big Sci* **4**, 8 (2020).

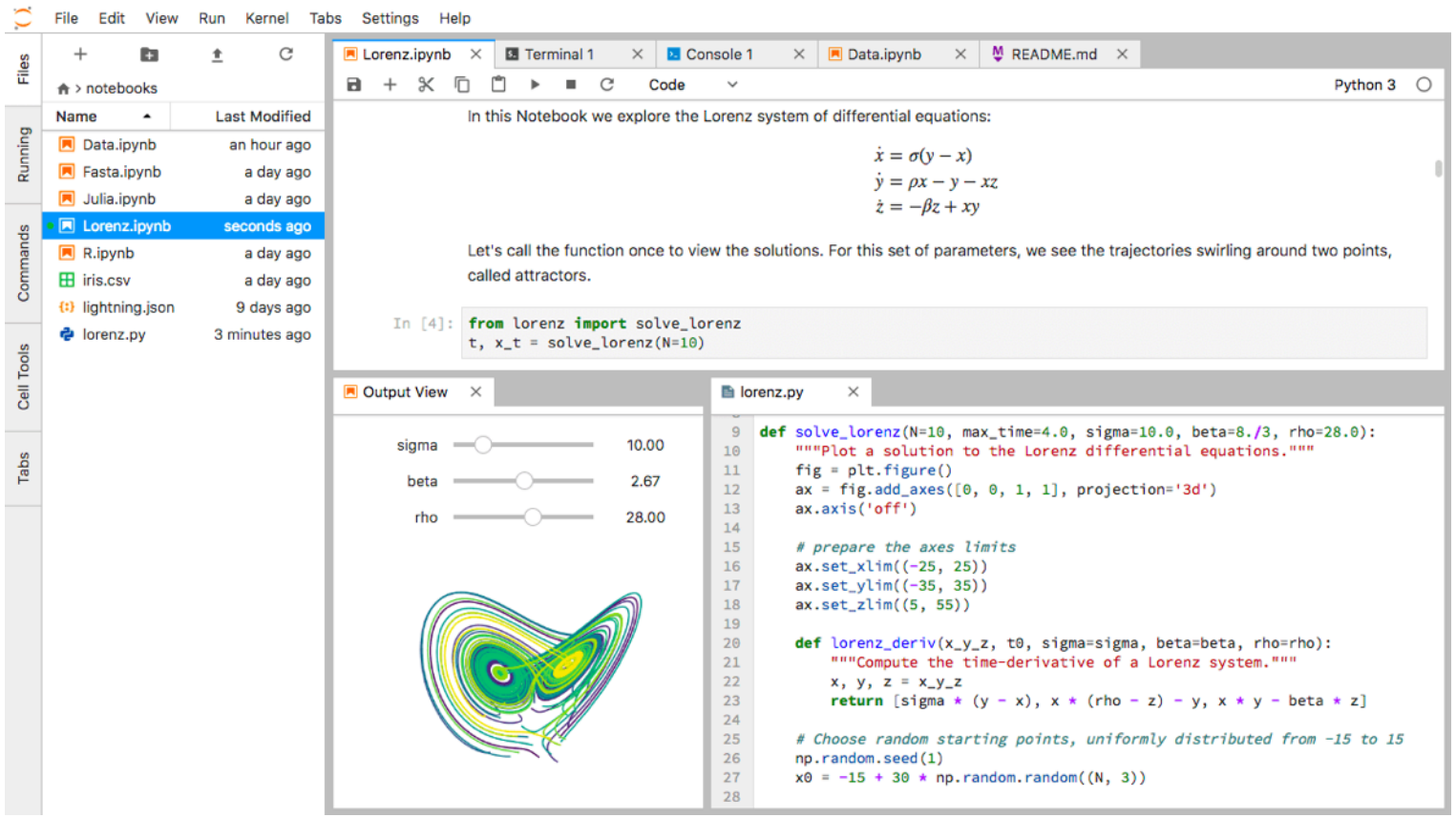
https://github.com/cerndb/SparkDLTrigger

CERN

Recent improvements and outlook

Jupyterlab on SWAN

- Next-generation interface for Project Jupyter
  - “IDE-like” environment
- Next steps: integration of current extensions
  - SWAN Projects
  - CERNBox sharing integration
  - Spark Connector and Monitor
  - …



SWAN on Kubernetes

- **We have refactored the backend of SWAN to run on Kubernetes**
  - Until now, notebooks have run in containers on physical servers
  - Kubernetes is a cluster manager for containerized applications
- **This allows to run SWAN in Cloud deployments**
  - SWAN runs in CERN cloud
  - Access GPUs from CERN cloud
- **Pilot projects to run in public clouds**
  - While accessing CERN storage (EOS) and software
  - Idea: overflow capacity in periods of high demand

SWAN on Kubernetes

- Currently, one pilot instance of SWAN on Kubernetes at https://swan-k8s.cern.ch
  - It will become the default instance at https://swan.cern.ch
- Through CERN openlab, a test cluster has been deployed on OCI (Oracle)
  - In the future, also other public cloud providers will be tested

GPUs for SWAN

- In the pilot instance in CERN cloud, we are offering 5 GPUs (4x Tesla T4 + 1x V100) as of October 2020
  - If there is demand, we will ask for more from CERN cloud
- How are the resources shared?
  - The user gets 1 GPU, 2 cores and 16 GB RAM from the available pool
  - Users are removed after 4 hours of inactivity
- Software packages from CVMFS
  - The latest release has Tensorflow 2.1.0 and PyTorch 1.4.0
  - The Bleeding Edge release has Tensorflow 2.3.0 and PyTorch 1.4.0
  - You can install your own with `pip --user install`

Demo

https://indico.cern.ch/event/852553/contributions/4060355/attachments/2125796/3579027/GPU_demo.mp4

- Thanks to: the SWAN Team (EP-SFT, IT-ST, IT-CM, IT-DB) and CERN openlab
- Thanks to all the users for the feedback they provide

- Access for beta users to GPUs at https://swan-k8s.cern.ch
  - Please contact us on Service Now to ask access to this cluster

CERN