SWAN
Service for
Web-based ANalysis

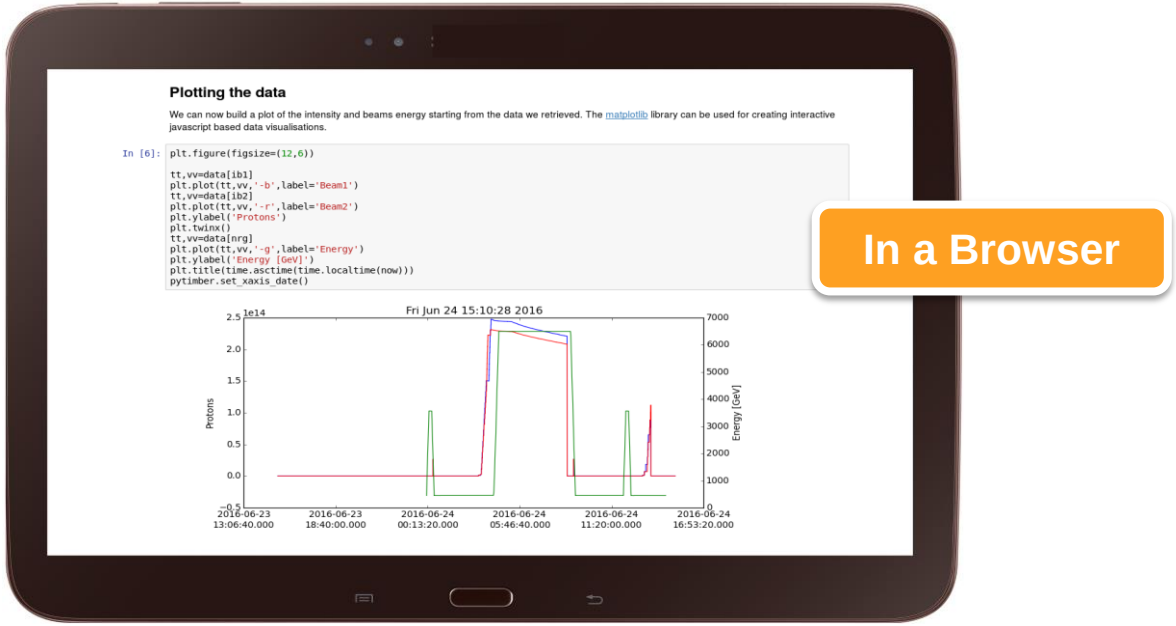
https://swan.web.cern.ch

E. Tejedor, D. Piparo, P. Mató
L. Mascetti, J. Moscicki, M. Lamanna
CERN

CHEP
12/10/2016

Interface: The Notebook

Jupyter Notebook: A web-based interactive computing interface and platform that combines code, equations, text and visualisations



Interface: The Notebook

Text

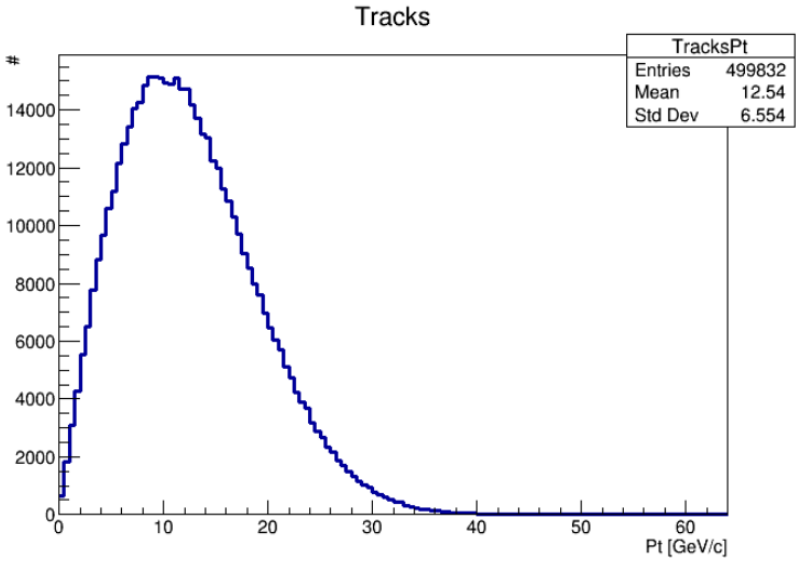
Code

Graphics

Access TTree in Python using PyROOT and fill a histogram

Loop over the TTree called "events" in a file located on the web. The tree is accessed with the dot operator. Same holds for the access to the branches: no need to set them up - they are just accessed by name, again with the dot operator.

In [1]:
```
import ROOT

f = ROOT.TFile.Open("http://indico.cern.ch/event/395198/material/0/0.root");
h = ROOT.TH1F("TracksPt","Tracks;Pt [GeV/c];#",128,0,64)
for event in f.events:
    for track in event.tracks:
        h.Fill(track.Pt())
c = ROOT.TCanvas()
h.Draw()
c.Draw()
```



Interface: The Notebook

Jupyter Notebook: A web-based interactive computing interface and platform that combines code, equations, text and visualisations

ROOT has been fully integrated with the Jupyter technology

- Two flavours: Python and ROOT C++
- JavaScript interactive visualisation

10/10,15:00 – *E. Tejedor*
The New ROOT Interface: Jupyter Notebooks

SWAN: Data analysis “as a service”

Interface: Jupyter Notebooks

Goals:

- Analysis only with a web browser
  - Platform independent ROOT-based data analysis
  - Calculations, input and results “in the Cloud”
- Easy sharing of scientific results: plots, data, code
  - Storage is crucial: mass & synchronised
- Simplify teaching of data processing and programming
  - Gallery of analysis examples
- Integration with other analysis ecosystems: R, Python, …

SWAN in the CERN Ecosystem

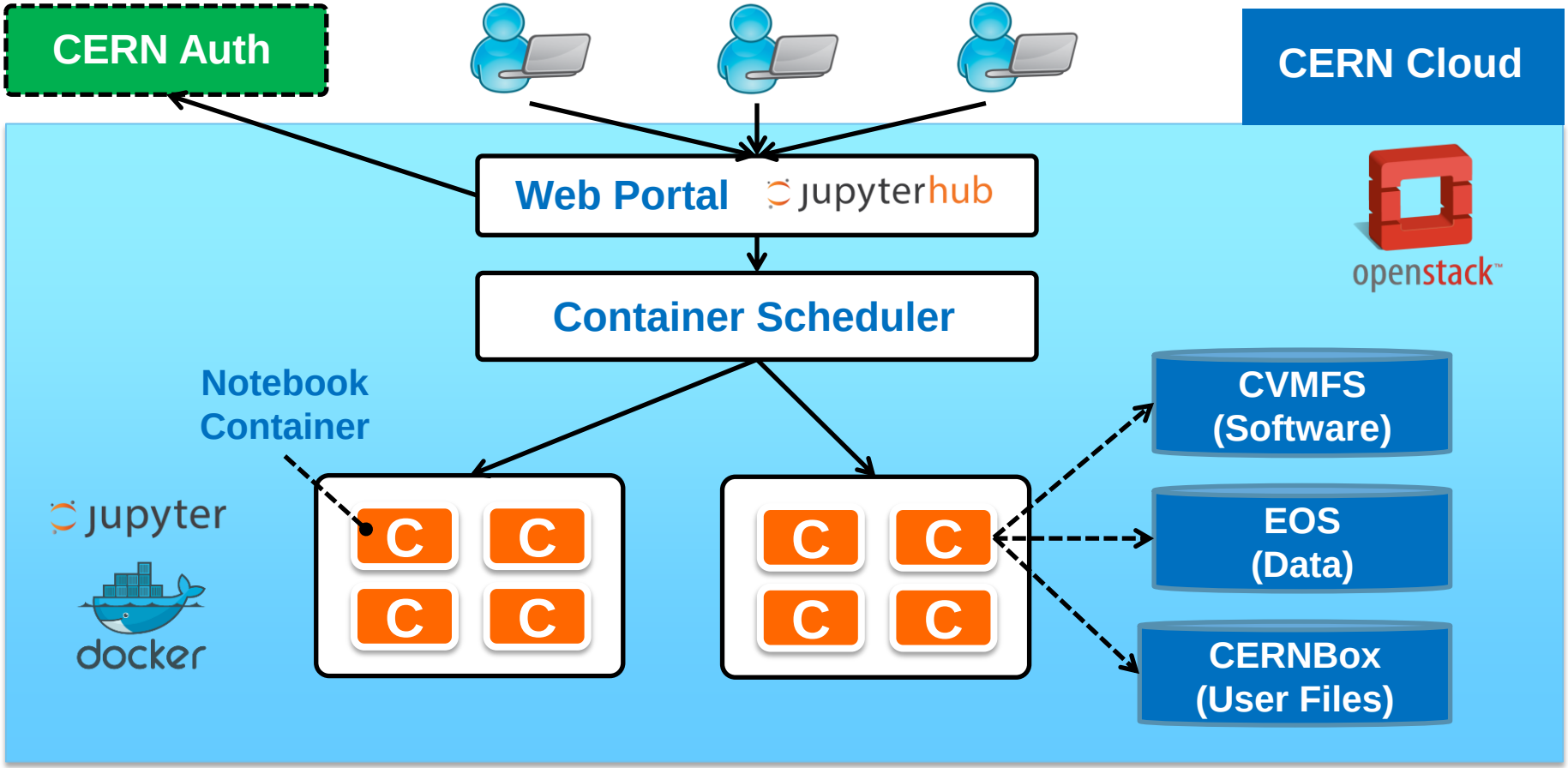
SWAN federates a set of production technologies at CERN:

- Authentication with CERN credentials (SSO)
- Infrastructure: virtual machines in OpenStack Cloud
- Software distribution: CVMFS
  - Centrally distributed software
- Storage access: CERNBox, EOS
  - Experiments' and users' data

Plus some external technologies:

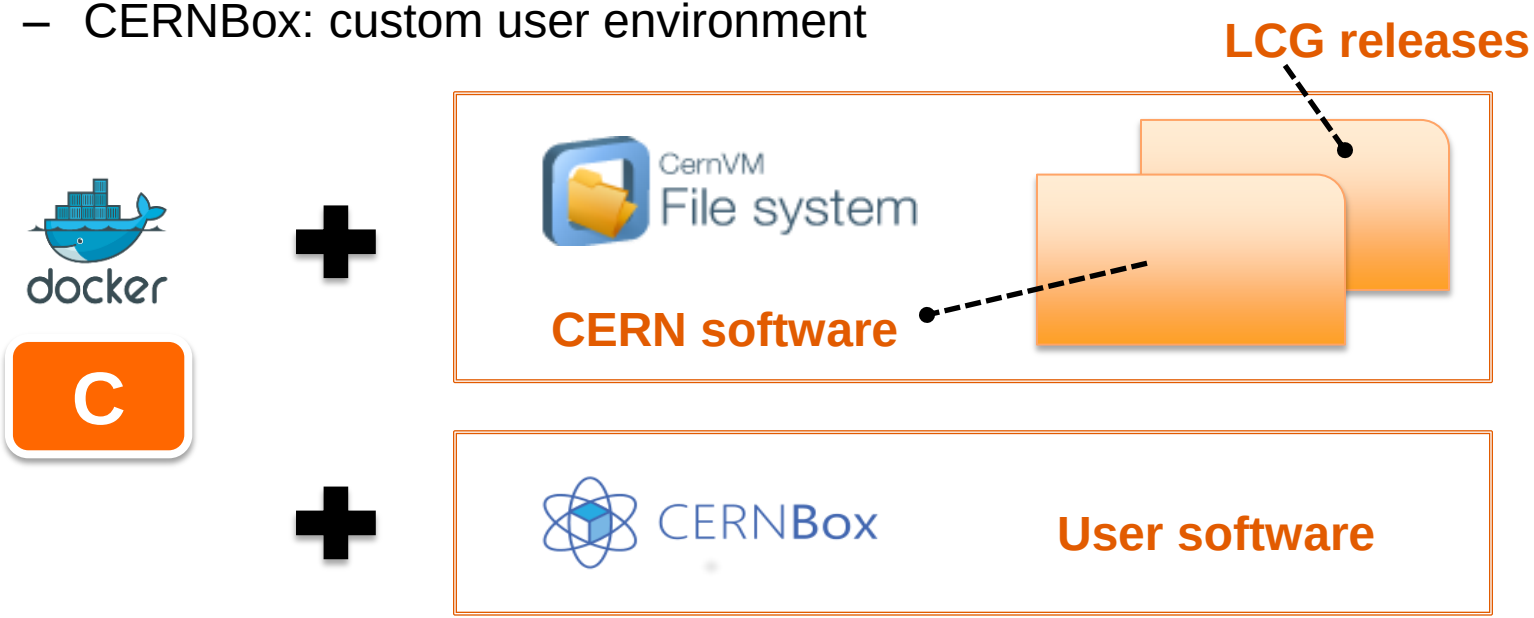
- JupyterHub
- Docker

Service Architecture

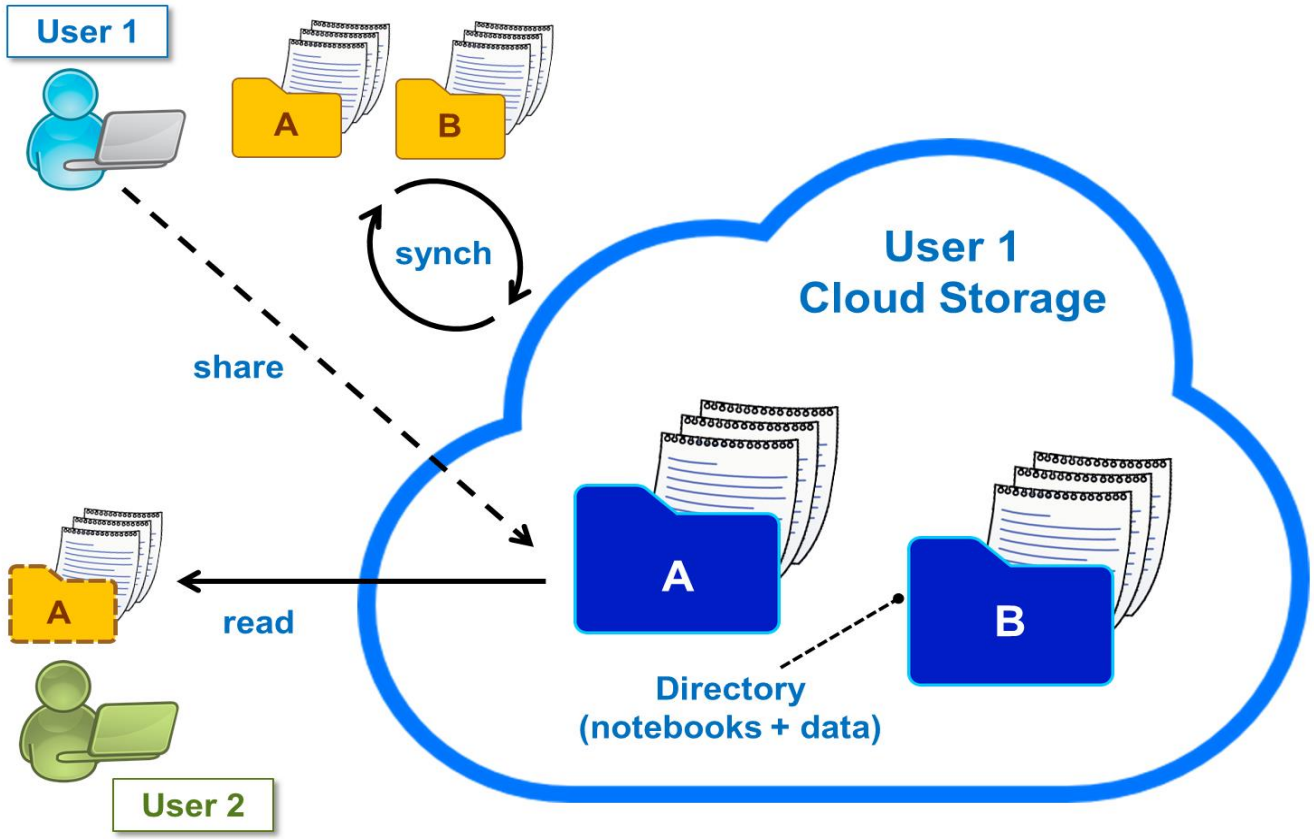


Software Environment

- Strategy to configure the software environment:
  - Docker: single thin image, not managed by the user!
  - CVMFS: configurable environment via “views”
  - CERNBox: custom user environment

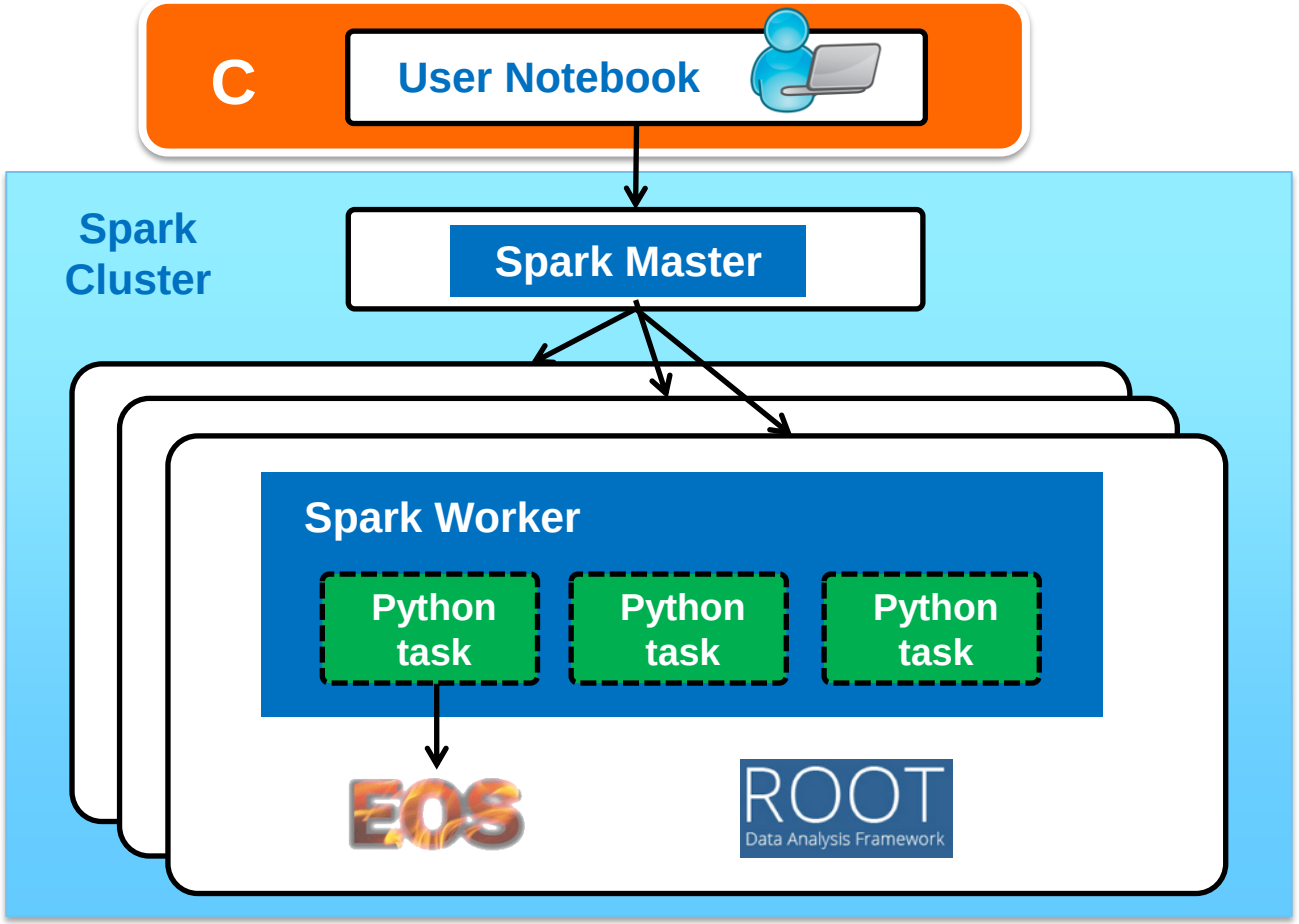


CERNBox: Sync & Share



11/10,16:30 –
J. Moscicki
CERNBox: The Data Hub for Data Analysis

R&D: Offloading from SWAN



In collaboration with IT-DB, IT-ST

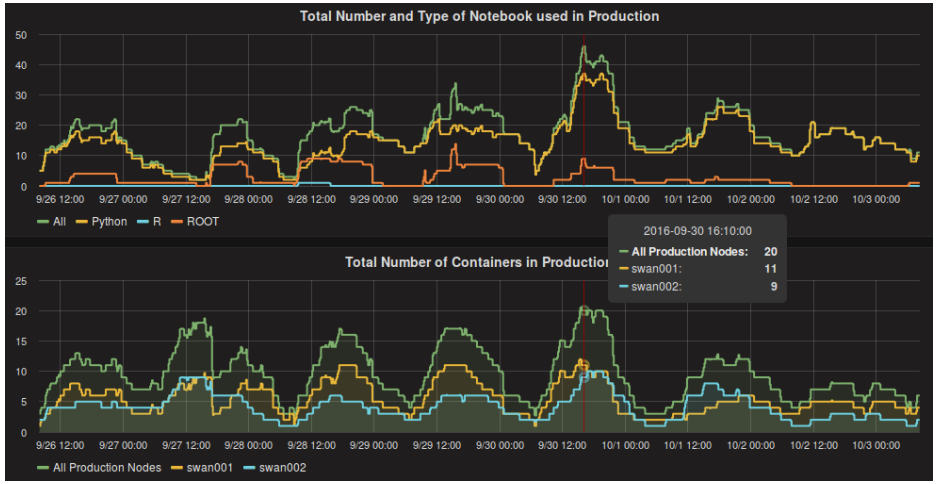
10/10,14:45 – *D. Piparo* Expressing Parallelism in ROOT

SWAN Today

- Pilot Service released in June

https://swan.cern.ch

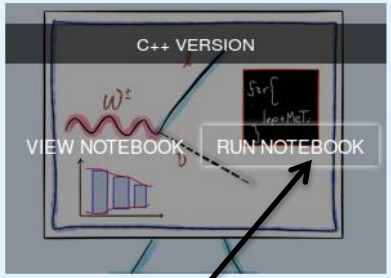
- Open to everyone with a CERN account
  - And a CERNBox account activated
- Stats of the first 100 days (Jun – Sep):
  - 1800 sessions created
  - 3700 notebooks opened
  - Peak of 100 sessions
- See backup slides for some user examples
  - E.g ATLAS Open Data, LHC Page 1 (BE)



ATLAS Open Data and SWAN

ATLAS ROOTbooks Gallery!

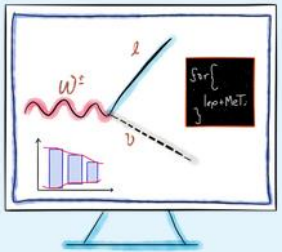
Analysis notebooks at
http://opendata.atlas.cern/webanalysis/ROOTbooks.php

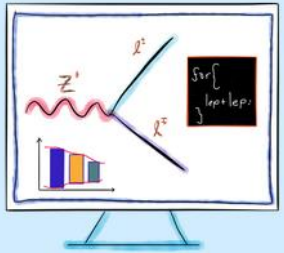


Runnable in SWAN!

The W Analysis ROOTbook

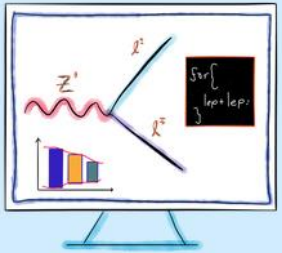
The W boson analysis is intended to provide an example for a high statistics analysis using the ATLAS open data dataset. Furthermore it tests the description of the real data by the simulated W boson data which represents the most extensive dataset in terms of luminosity.





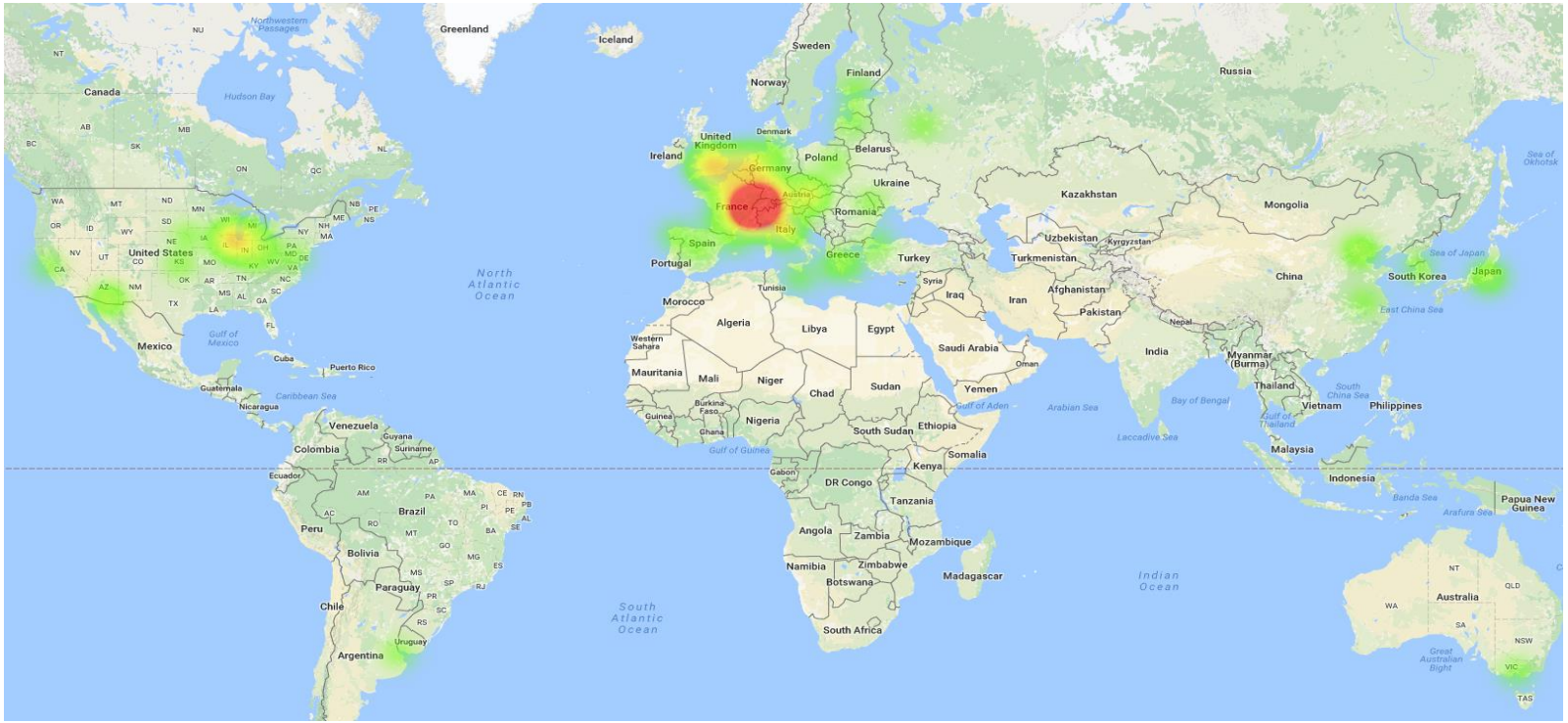
The Z Analysis ROOTbook

Many analyses selecting leptons suffer from Z + jets as a contributing background due to its large production cross section. It is therefore vital to check the correct modelling of this process by the Monte-Carlo simulated data. It is important to measure well known Standard Model particles, to confirm that we understand properly the detector and software. We are then ready to search for new physics.



SWAN Users

- Service open outside CERN since end of July

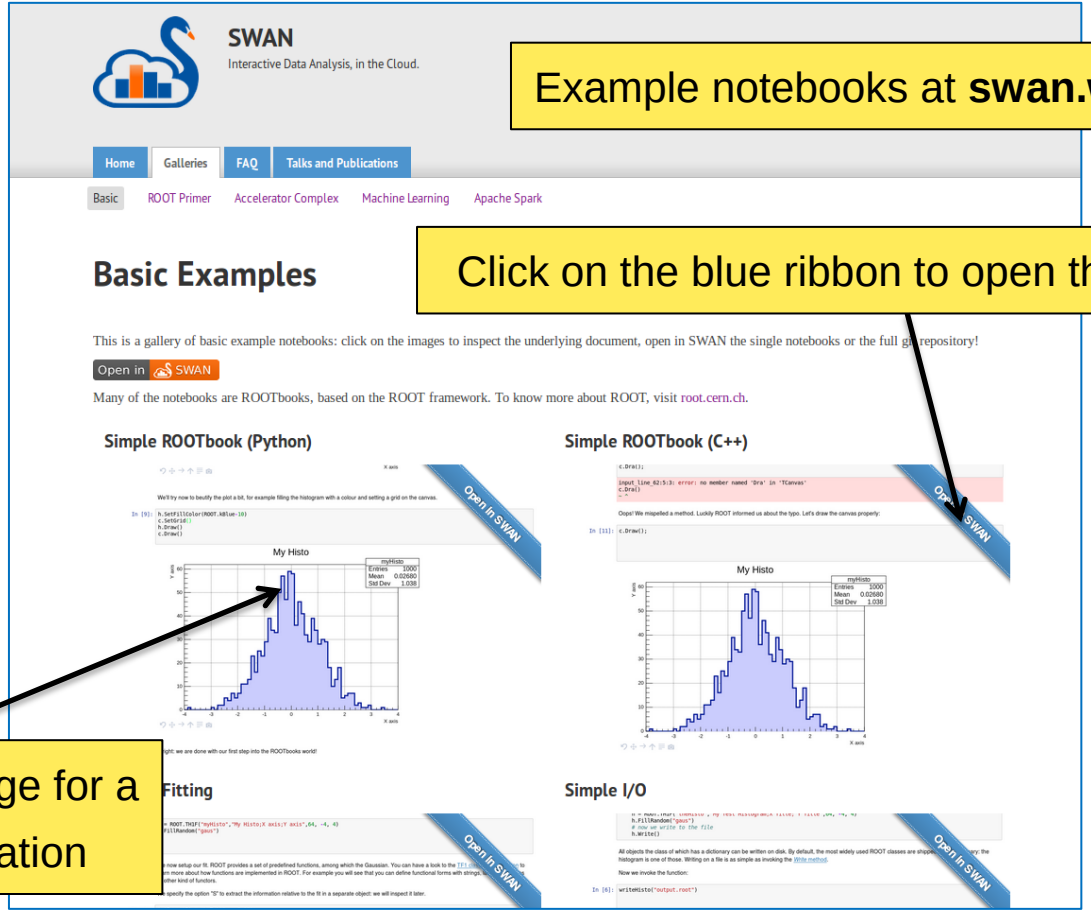


Summary, Plans

- SWAN pilot service available
  - ROOT integrated with Jupyter
  - CVMFS for software distribution
  - EOS mass storage + CERNBox synchronisation
- Future plans:
  - Incorporate user feedback
  - Facilitate use of experiment software
  - Enrich CVMFS repositories with new packages (e.g. ML libraries)
  - Improve experience with storage: sharing
  - Exploit external resources (e.g. Spark clusters)

Backup

A Notebook Gallery



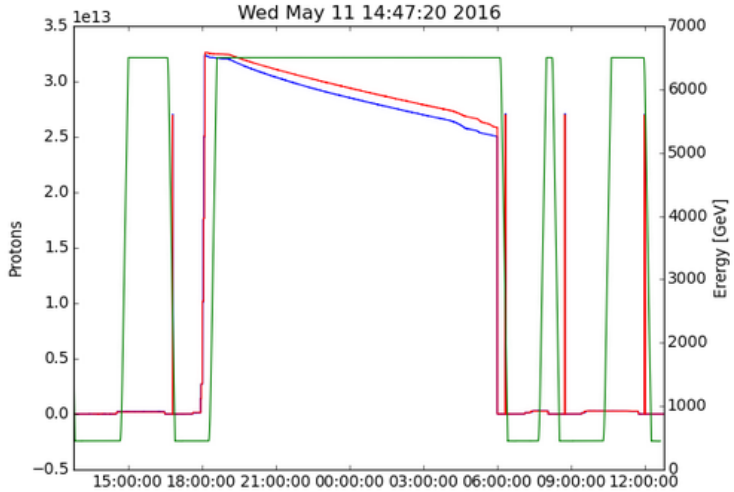
Example notebooks at **swan.web.cern.ch**

Click on the blue ribbon to open them in SWAN!

Click on the image for a static visualisation

```
In [3]: now=time.time()
        ib1="LHC.BCTDC.A6R4.B1:BEAM_INTENSITY"
        ib2="LHC.BCTDC.A6R4.B2:BEAM_INTENSITY"
        nrg="LHC.BOFSU:OFSU_ENERGY"
        data=db.get([ib1,ib2,nrg],now-3600*24,now)
```

```
In [4]: plt.clf()
        tt,vv=data[ib1]
        plt.plot_date(epoch2num(tt),vv,'-b',label='Beam1')
        tt,vv=data[ib2]
        plt.plot_date(epoch2num(tt),vv,'-r',label='Beam2')
        plt.ylabel('Protons')
        plt.twinx()
        tt,vv=data[nrg]
        plt.plot_date(epoch2num(tt),vv,'-g',label='Energy')
        plt.ylabel('Erergy [GeV]')
        plt.title(time.asctime(time.localtime(now)))
```



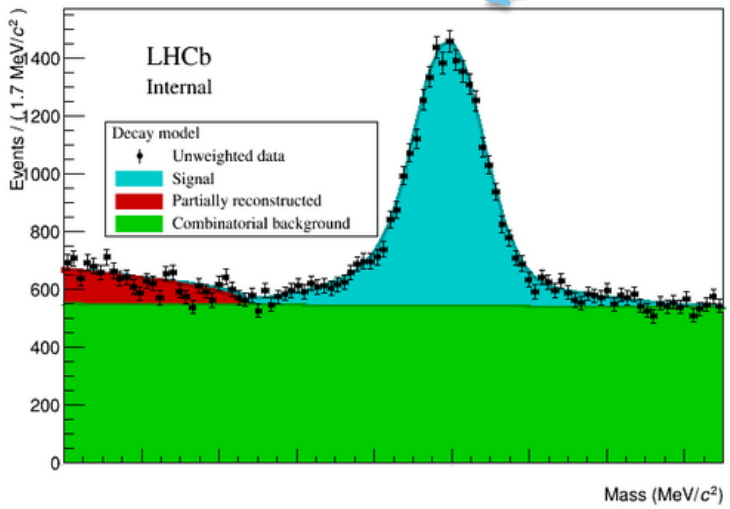
R. De Maria, BE-ABP-HSS

https://github.com/rdemaria/pytimber/blob/master/examples/LHC%20Page1.ipynb

- Read measurements coming from pick-ups in a database
- Plot time series
- Needs also SciPy and to share the notebooks with his colleagues

```
title = { "model": "Signal" ,  "pdfBkg" : "Partially reconstructed" , "cmbBkg": "Combinatorial background"}

for (component, color) in [ ("model",kCyan), ("pdfBkg",kRed), ("cmbBkg",kGreen)]:
    model.plotOn (frame, LineColor(color+2) , DrawOption('L'), Components(component), LineWidth(5))
    model.plotOn (frame, FillColor(color+1) , DrawOption('F'), Components(component), LineWidth(0), Name("P"+component
))
    leg.AddEntry ( frame.findObject ("P"+component), title[component] , "F" )

data.plotOn ( frame, MarkerColor ( ROOT.kBlack ) )
frame.Draw()
Graphics().lhcbMarker(0.2,0.8, "Internal")

leg.Draw()

ROOT.gPad.Draw()
```

Results coming from real data! (published now)



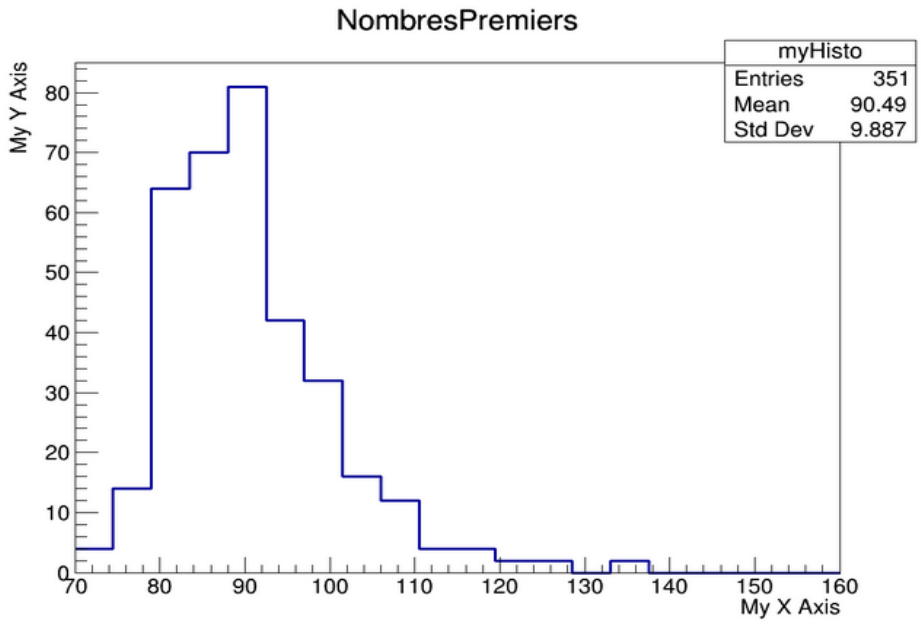
L. Anderlini

Rare B meson decay in LHCb

- Read data from EOS
- Setup complex fit
- Document and inspect results

```
In [138]: import ROOT
htemp = ROOT.TH1F("myHisto","NombresPremiers;My X Axis;My Y Axis",20,70,160)
for i in range(len(data)):
    d = data[i][0]
    htemp.Fill(float(d))
c = ROOT.TCanvas("myCanvas","myCanvasTitle",1024,768)
htemp.Draw()
c.Draw()
```

```
TROOT::Append:0: RuntimeWarning: Replacing existing TH1: myHisto (Potential memory leak).
TCanvas::Constructor:0: RuntimeWarning: Deleting canvas with same name: myCanvas
```



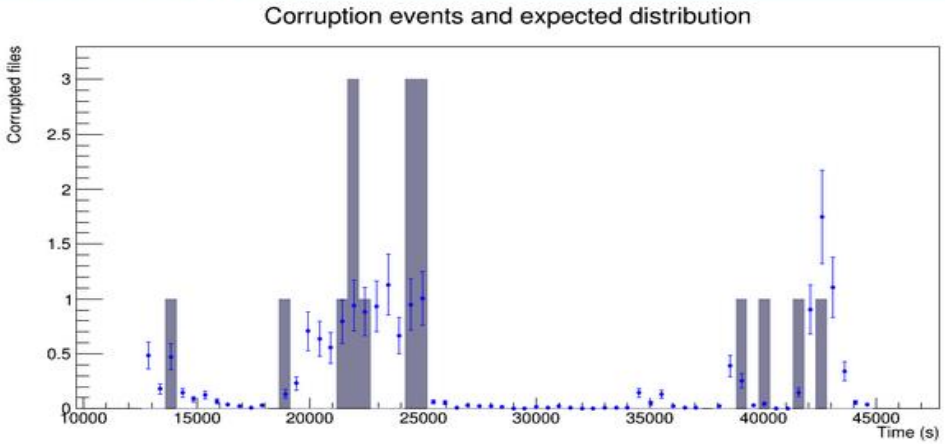
Mano S. (14 years old), K12 student

- Approaches programming for the first time
- Verifies numerically what he learned at school
- Shares results with his supervisor and classmates

```
Graph_x.Draw("P")

# Predefined function:
#fit_x1 = ROOT.TF1("fit_x1", "pol2", 0, 900)
#fit_x1.SetLineColor(kRed)
#Graph_x.Fit("fit_x1")

# Writing function explicitely:
#myfit = ROOT.TF1("myfit", "[0] + [1]*x", 0, maxt-mint)
#myfit.SetParameters(0,10.0)
#myfit.SetParameters(1,0.0022)

#myfit.SetLineColor(kRed)
#Graph_x.Fit("myfit","+")        # Option "+" to add fit

canvas.Update()
canvas.Draw()
if (SavePlots):
    canvas.SaveAs("Figure2.png")
```

```
Chi2/ndf with the hyphotesis corruption prob = 0.0022: 55.43/62 in the interval from Wed Jun 10 23:30:00 2015 to Thu
Jun 11 08:30:00 2015
```

```
Info in <TCanvas::Print>: png file Figure2.png has been created
```



G. Lo Presti, M. Lamanna

"Castor data corruption incident"

- Describe incident, data source, analysis and results in a single document