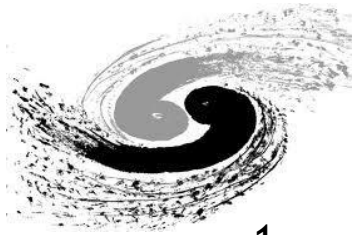
Synchronization and Sharing Services in IHEP

Qi Mengyao ,Lin Tao
On behalf of Computer Center, IHEP
qmy@ihep.ac.cn



Contents

- *IHEP Introduction*
- *IHEPBox at IHEP*
- *Data Analysis Service*
- *Ongoing work*
- *Summary, Plans*

Major Experiments at IHEP

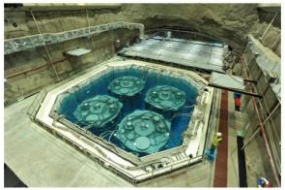
- IHEP: The largest fundamental research center in China
- IHEP serves as the backbone of China's large science facilities

BEPC & BEPCII

BES III
Beijing Spectrometer III

Beijing Synchrotron
Radiation Facility

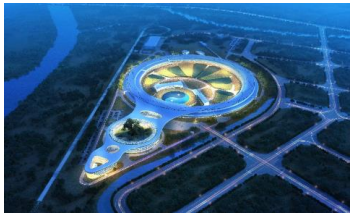
Yangbajing Cosmic
Ray Observatory



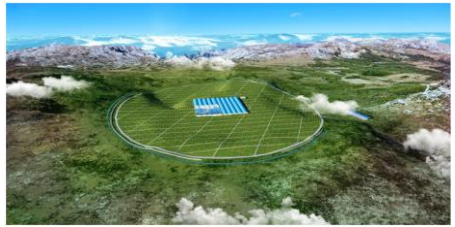
Daya Bay Reactor
Neutrino experiment

HXMT
Hard X-ray
Modulation Telescope

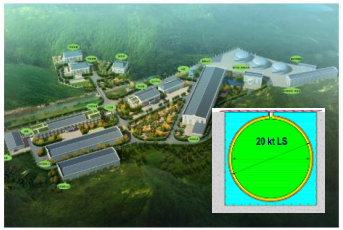
CSNS
China Spallation
Neutron Source



HEPS
High Energy
Photon Source



LHAASO
Large High Altitude Air
Shower Observatory



JUNO
Jiangmen Underground
Neutrino Observatory

Major Experiments at IHEP

- BEPCII/BESIII
  - 5PB data in 5 years
- DYB
  - 400TB per year data collected
- JUNO
  - ~2PB raw data per year

lustre

- **LHAASO**
  - ~6PB raw data per year
  - accumulate 20PB+ in 10 years
- HXMT

EOS

- Atlas and CMS Tier2 site
  - 940TB disk, 1088 CPU cores

IHEPBox at IHEP

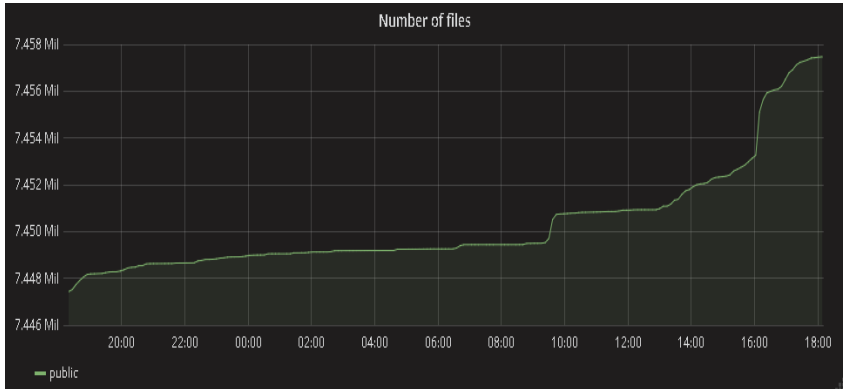
IHEPBox provide a cloud synchronization service

- Available for all IHEP users
- Synchronize files (data at IHEP) and data access
- Easy way to share with other users
- All major platforms supported
- Based on ownCloud integrated with EOS

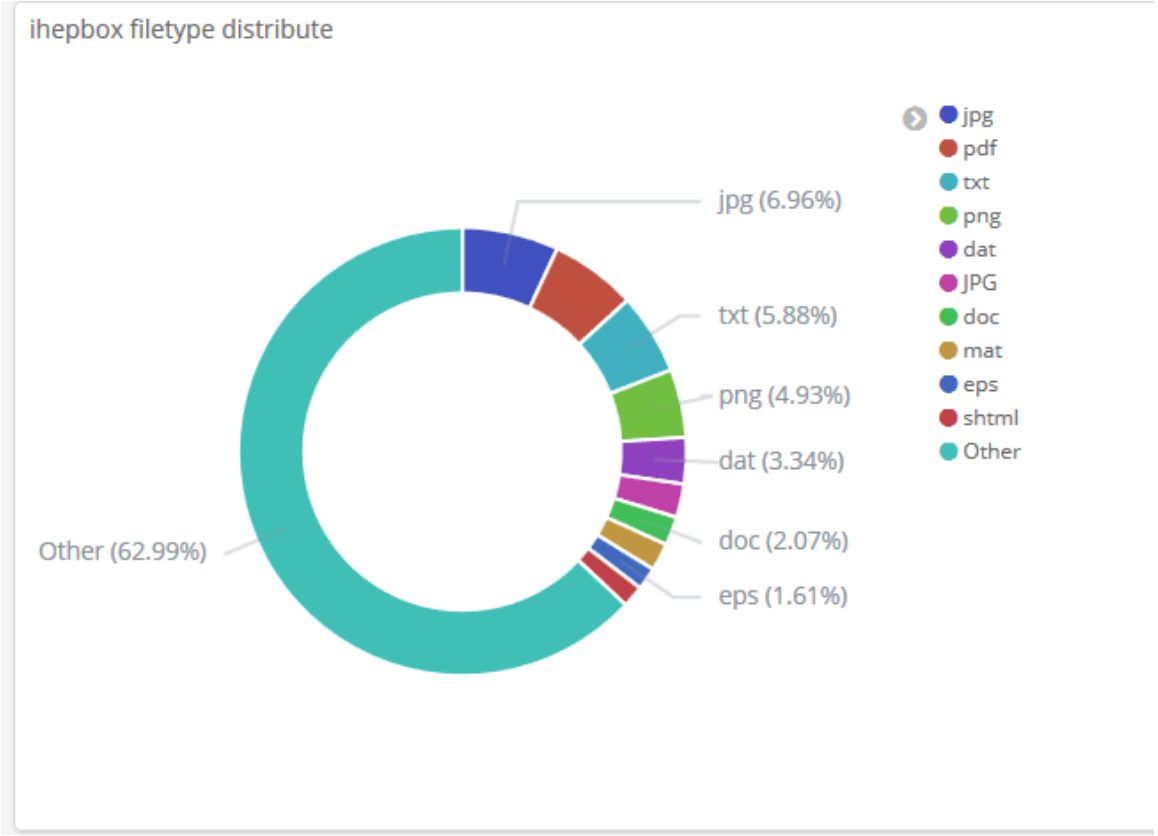
IHEPBox Service Numbers

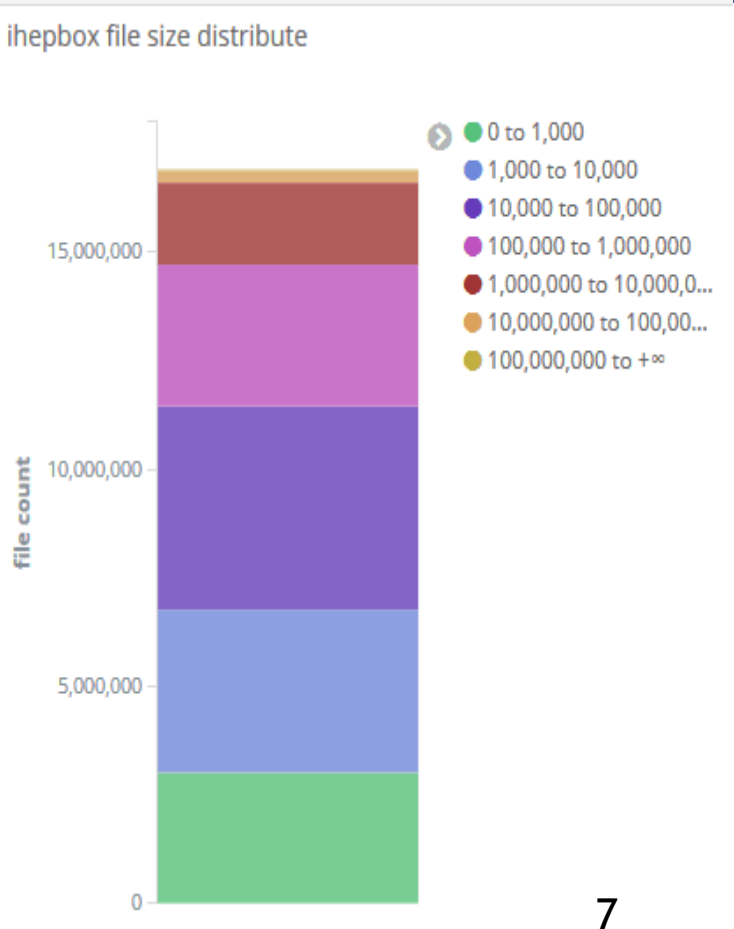
Fast growing service

- Add-ons to integrate new ways to interact with IHEPBox , such as documents/galleries/calendar/activity…
- Large and growing user base
- 8K+ users, 100GB per user
- 160 TB ,used 42TB
- 17M+ files



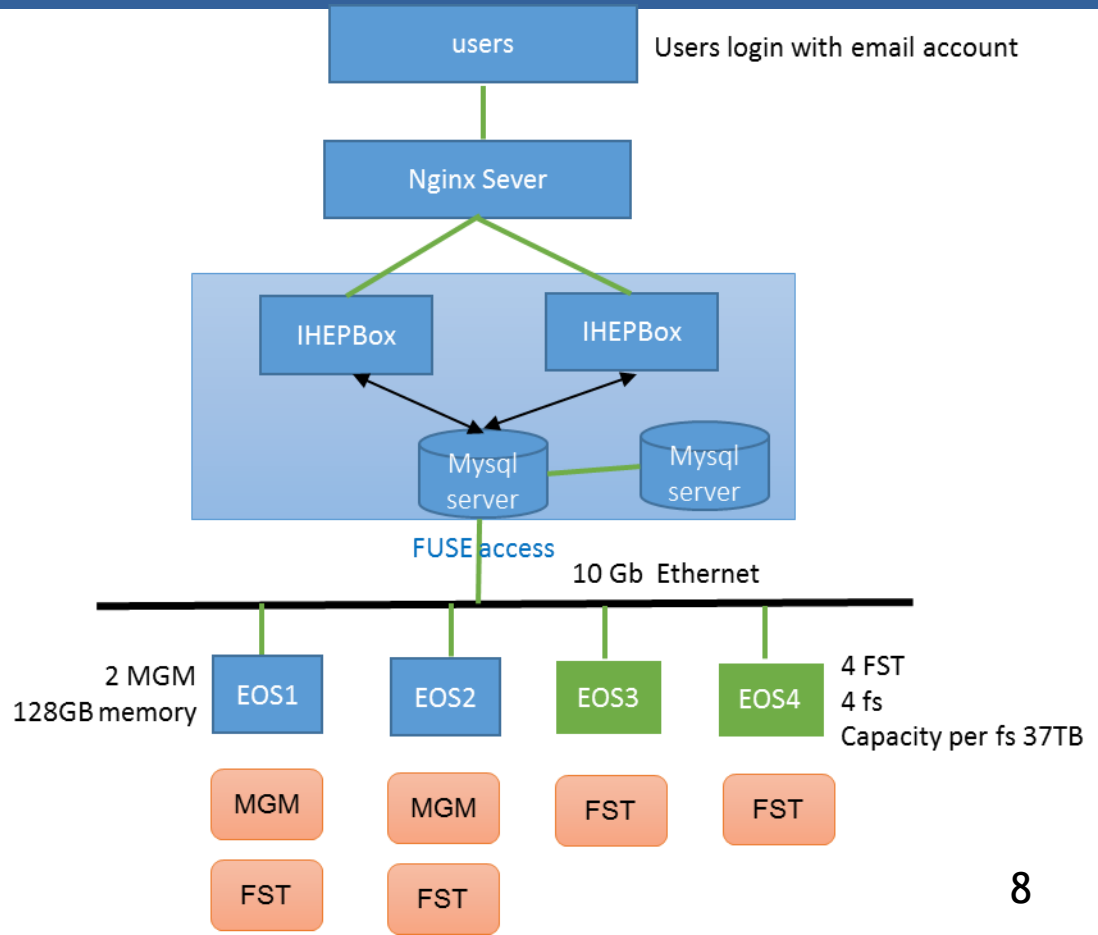
IHEPBox Service Numbers





IHEPBox at IHEP

- Users
  - IHEP LDAP users
  - Local users
- Application Service
  - 1 * Nginx
  - 2 * IHEPBOX Servers
- Data Storage
  - 4 * fst
  - Two replication
  - 2 * Mysql servers

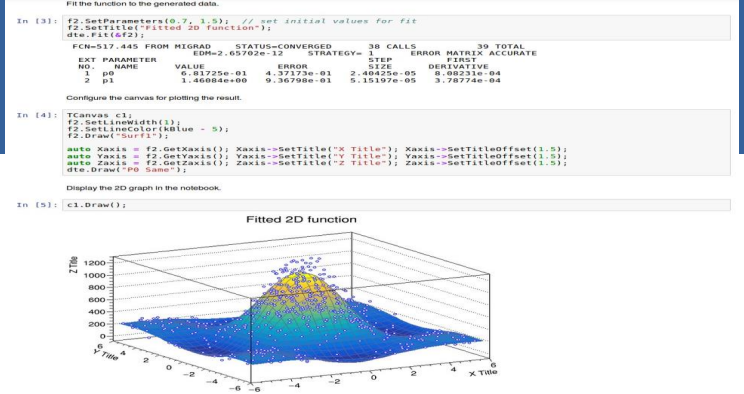


Data analysis service

- Physics analysis tasks is challenge due to “big data” (PB scale).
- A physics analysis task is split into many jobs manually, while each job will process a part of dataset.
- For the physics analysis, the same dataset will be loaded many times due to iteration.
  1、Write code, edit job option and update selection criteria.
  2、Submit batch jobs.
  3、After jobs finished, plot data, analyze result.
  4、Go to step 1.
- Submitting jobs and plotting are the most time consuming part.
- Loading dataset many times also cause I/O performance.

Q: How to reduce the iteration time? A: “**Analysis as a Service**”

Analysis workflow

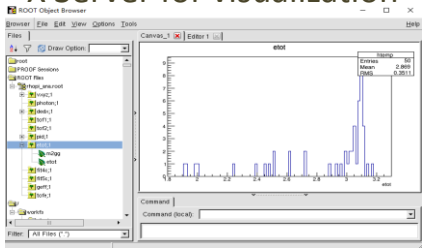


Web browser

AFS/SSO Auth

SSH client

X Server for visualization



Login Node

Submit batch jobs

HTCondor Scheduler

web-based analysis service

- Common software
- Middleware
- Computing facilities

Powerful computing resources

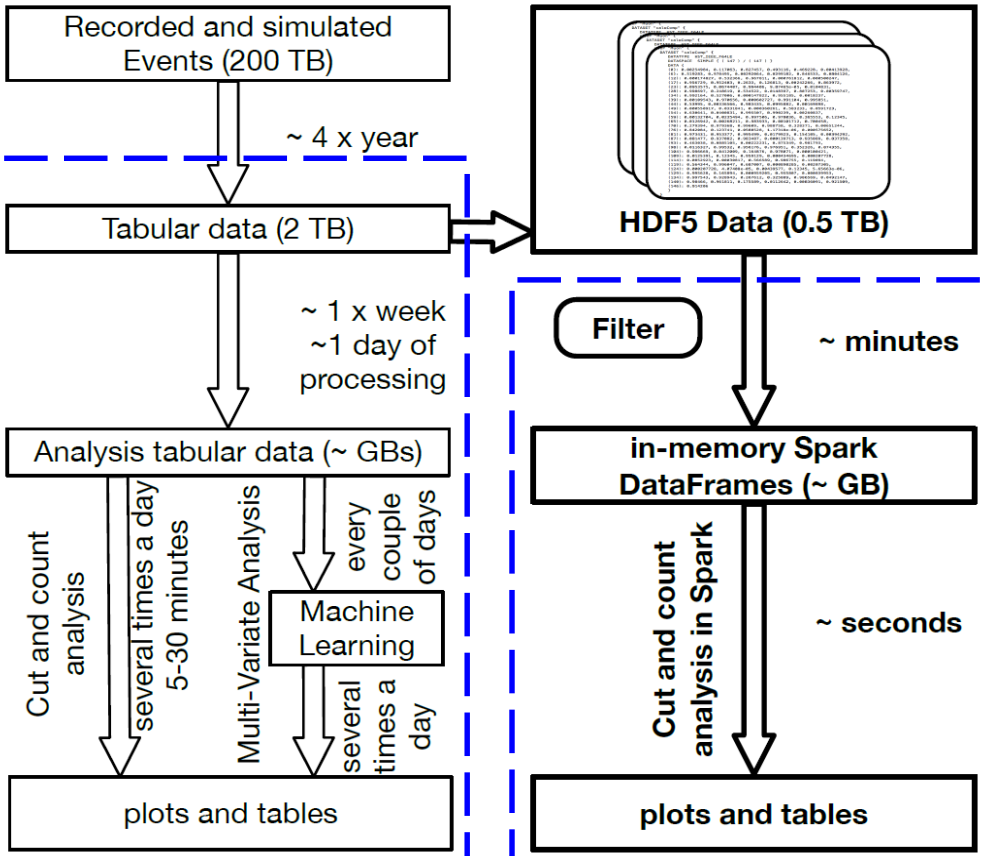
/afs

/besfs

Storage

Computing Node

Using “big data” technology: in-memory analysis. Read once, Analysis multiple times



Traditional analysis

- Event by event processing.
- Split by jobs, then merge results.
- Intermediate files are created for each stage.
- I/O bottleneck.

Data structure become more important.

“big data” analysis

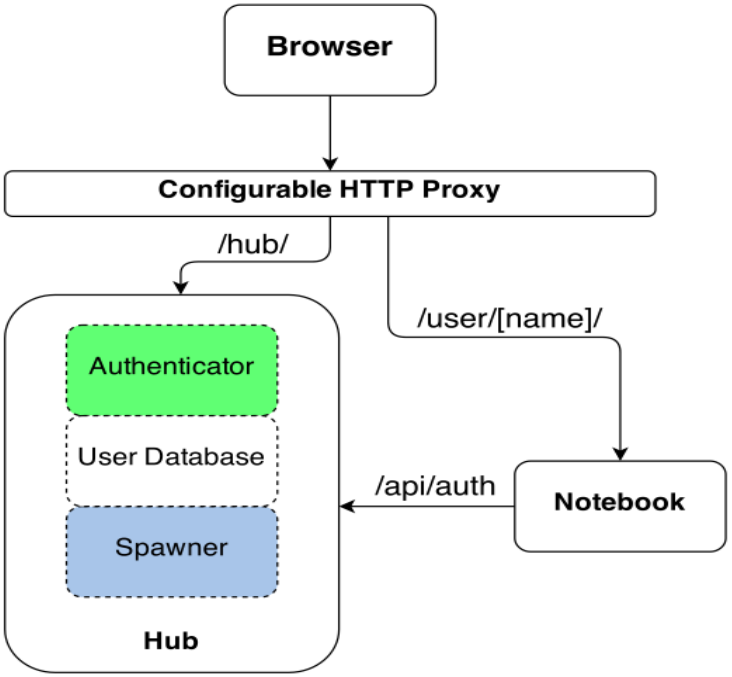
- In memory analysis.
- The properties are processed parallel.
- Join operation later.

RDataFrame in ROOT: https://root.cern.ch/rdataframe-session-chep-2018-0

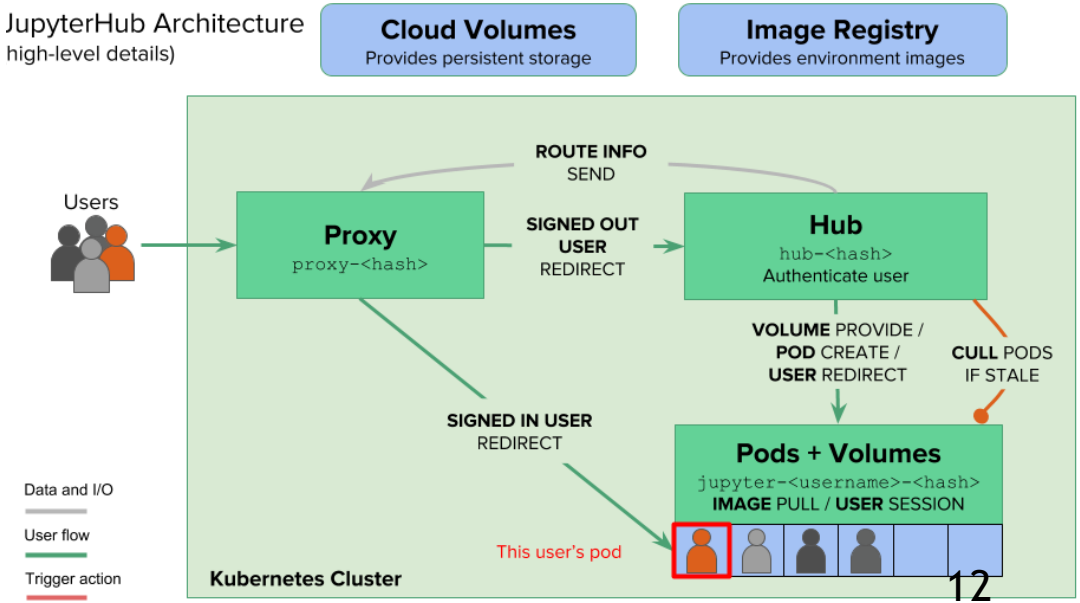
Architecture:

+

kubernetes

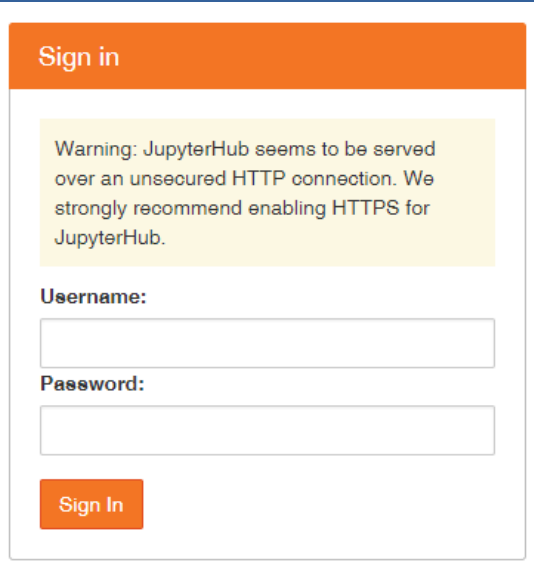


- Kubernetes manages the computing resources.
- On-demand, scalable computing.
- The backend can be also integrated with HTCondor.



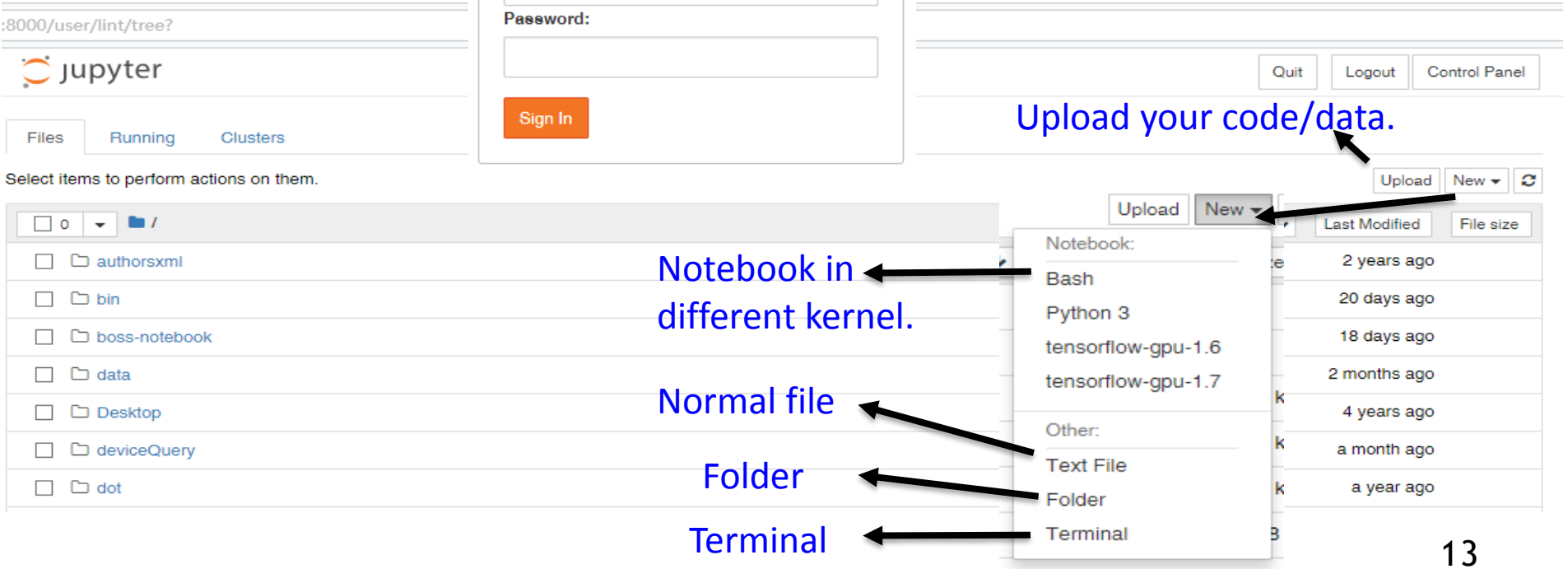
- JupyterHub manages multiple instances of the single-user Jupyter notebook.
- Authenticates users.
- Spawns notebooks for users.

Login using AFS account.



Current AFS status

- 2521 IHEP AFS users



Fit the function to the generated data.

In [3]:
```
f2.SetParameters(0.7, 1.5);  // set initial values for fit
f2.SetTitle("Fitted 2D function");
dte.Fit(&f2);
```

```
 FCN=517.445 FROM MIGRAD    STATUS=CONVERGED      38 CALLS          39 TOTAL
                     EDM=2.65702e-12    STRATEGY= 1      ERROR MATRIX ACCURATE
  EXT PARAMETER                                   STEP         FIRST
  NO.   NAME      VALUE            ERROR          SIZE      DERIVATIVE
   1  p0           6.81725e-01   4.37173e-01   2.40425e-05   8.08231e-04
   2  p1           1.46084e+00   9.36798e-01   5.15197e-05   3.78774e-04
```

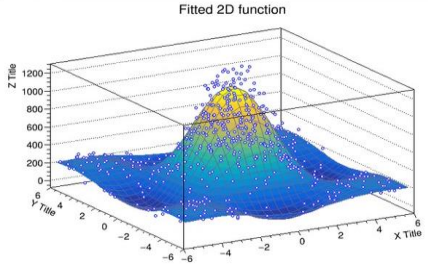
Configure the canvas for plotting the result.

In [4]:
```
TCanvas c1;
f2.SetLineWidth(1);
f2.SetLineColor(kBlue - 5);
f2.Draw("Surf1");

auto Xaxis = f2.GetXaxis(); Xaxis->SetTitle("X Title"); Xaxis->SetTitleOffset(1.5);
auto Yaxis = f2.GetYaxis(); Yaxis->SetTitle("Y Title"); Yaxis->SetTitleOffset(1.5);
auto Zaxis = f2.GetZaxis(); Zaxis->SetTitle("Z Title"); Zaxis->SetTitleOffset(1.5);
dte.Draw("P0 Same");
```

Display the 2D graph in the notebook.

In [5]:
```
c1.Draw();
```



Access TTree in Python using PyROOT and fill a histogram

First import the ROOT Python module.

In [1]:
```
import ROOT
%jsroot on
```

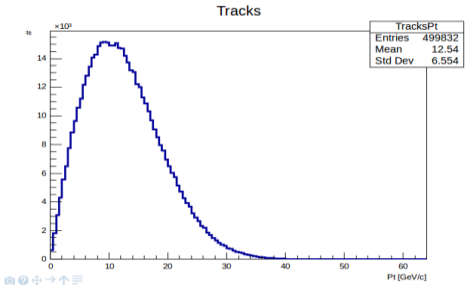
```
Welcome to JupyROOT 6.07/07
```

Open a file which is located on the web. No type is to be specified for "f".

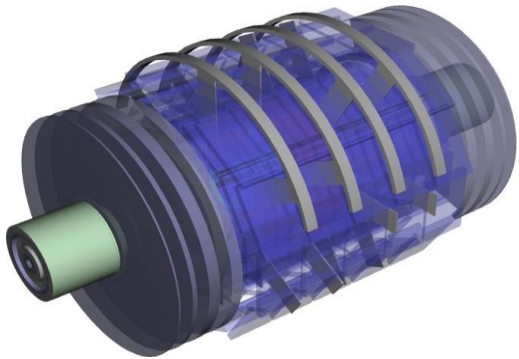
In [3]:
```
f = ROOT.TFile.Open("http://indico.cern.ch/event/395198/material/0/0.root");
```

Loop over the TTree called "events" in the file. It 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 [4]:
```
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()
```



Multiple language support: C++ and Python



ROOT Kernel extension.

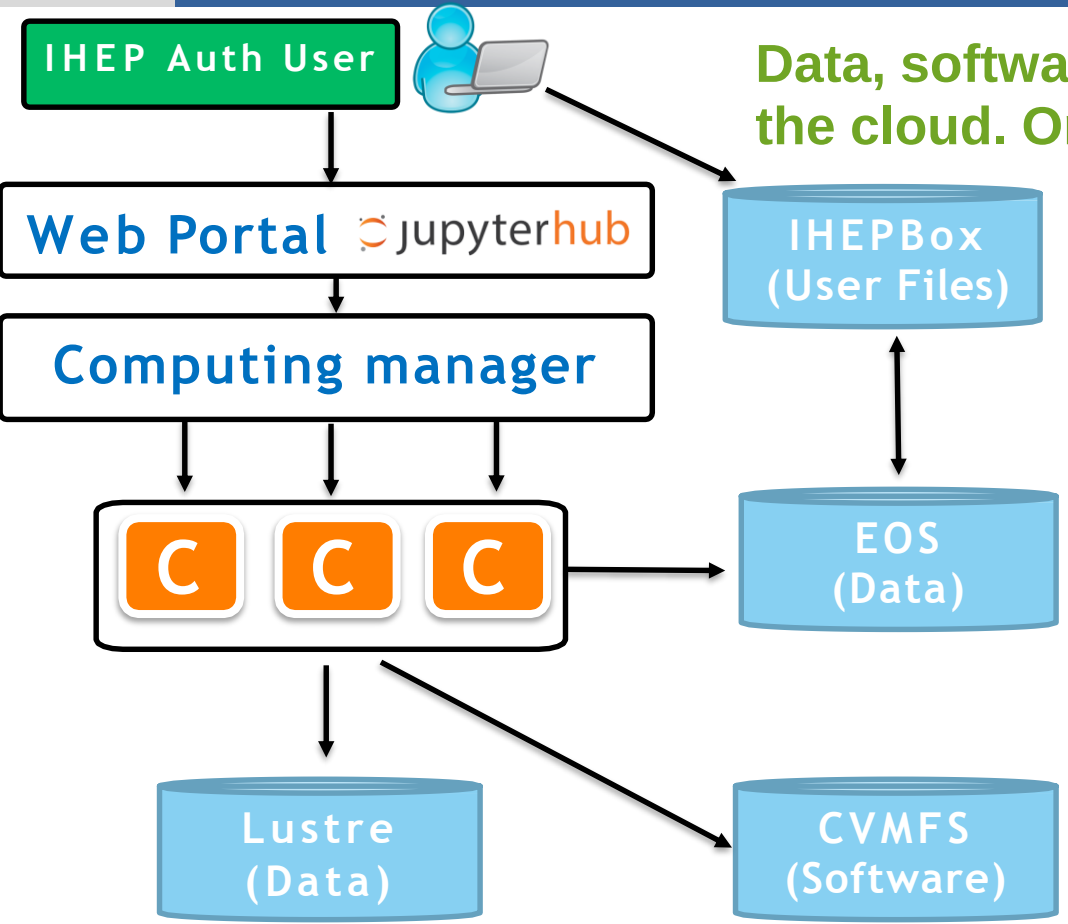
Status

- Jupyter software stack is deployed at AFS.
- Users can start Jupyter and ROOT 6 in their own space.
- Setup JupyterHub in a virtual machine and setup Kubernetes in two blade servers.

Challenges

- Need to klog manually to get AFS token.
- Support multiple users and experiments.
- Unified data access between SSH and Web.
- How to benefit from “big data” technology.

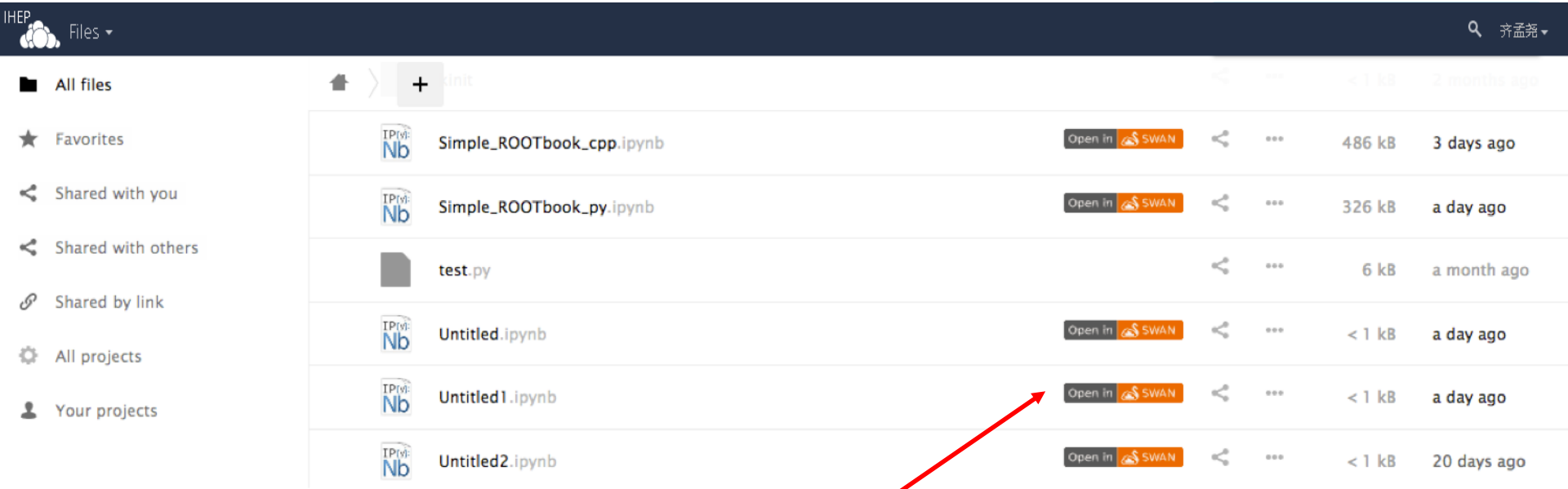
Ongoing： Cloud-based open platform



Data, software and computing resources in the cloud. Only a web browser is needed!

- Authentication： **IHEP SSO**
- Infrastructure:
  - **Jupyter + IHEPBox + EOS**
- Software distribution: **CVMFS**
- Storage, in two flavours:
  - DATA： EOS and Lustre
  - User Files: IHEPBox

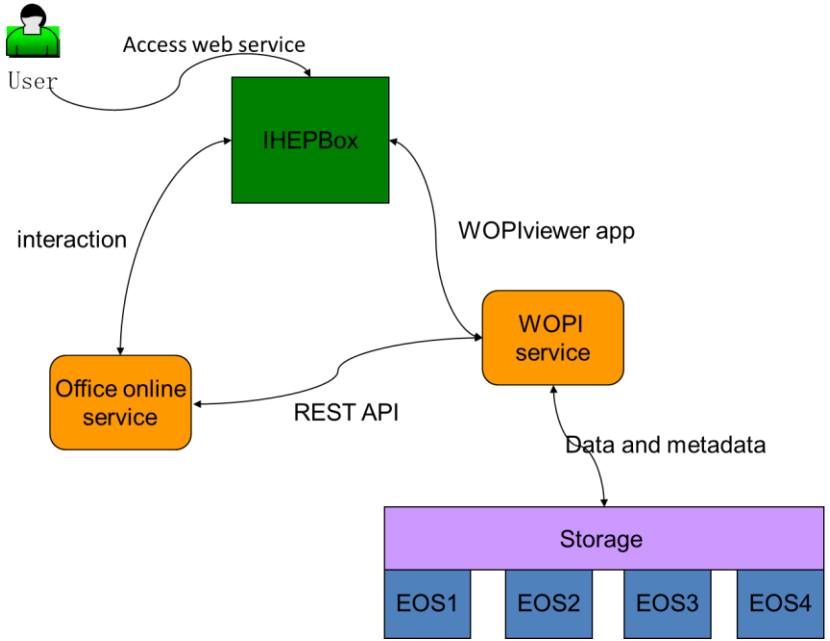
Ongoing： IHEPBox Integration

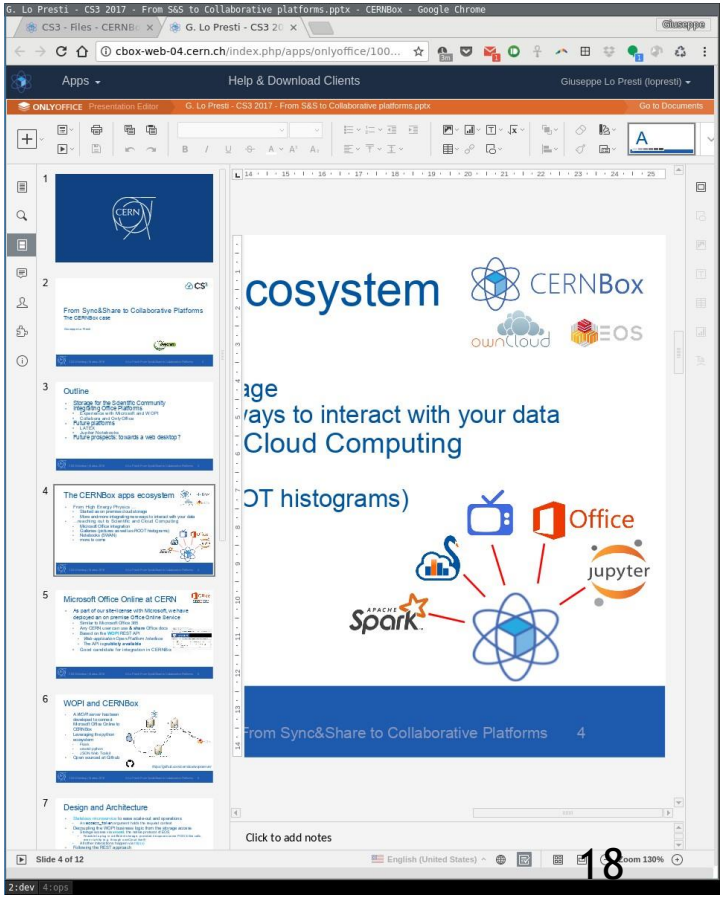


Notebook can be opened in IHEPBox

Ongoing： IHEPBox and WOPI server

A *WOPI server will* be developed to connect Microsoft Office Online to IHEPBox





Summary , Plans

- Prototype service available
  - CVMFS for software distribution
  - EOS、Lustre mass storage
  - IHEPBox Synchronization and sharing
- Future plans:
  - Jupyter service for deployment ,integrate with IHEPBox
  - Unified login accounts
  - Experiments data, users' data accessible on EOS
  - Office online service available
  - Integrate with experiments software
  - Open to all users