VO-CLOUD upgrade
Integration of Spark, Jupyter and HDFS in a UWS-driven cloud service

Petr Škoda, Jakub Koza

Astronomical Institute Academy of Sciences
Ondřejov
Czech Republic

Supported by grant COST LD-15113
of the Ministry of Education, Youth and Sports
of the Czech Republic

And

National Science Foundation of China

IVOA Interoperability meeting , GWS Session 1
Shanghai, China, 16th May 2017

Concept of scientific „CLOUD“

ITERATIVE REPEATING of SAME computation (workflow)

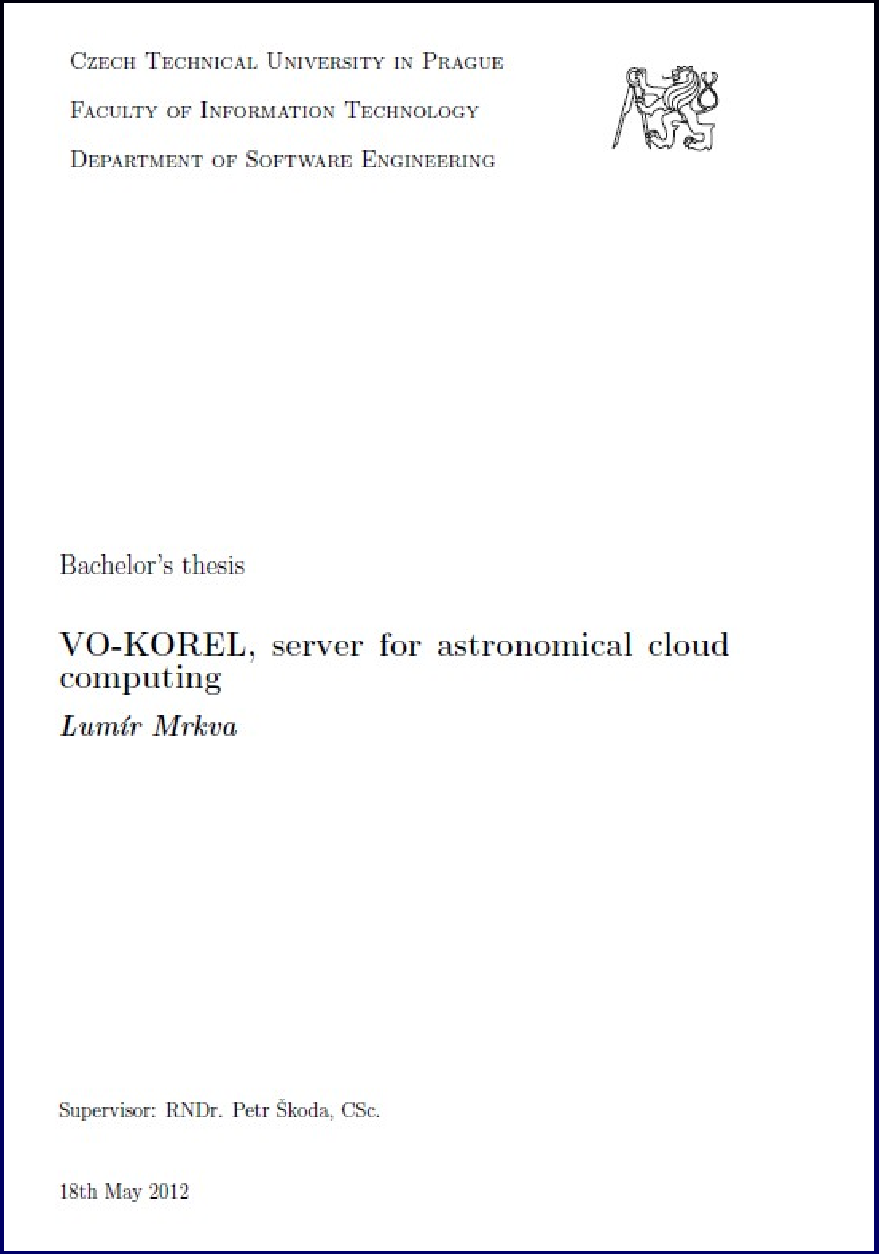
Machine Learning (of emission line profiles of LAMOST)

LARGE stable INPUT data + small changing PARAMS

Many runs on SAME data (tuning required)

Graphics visualization from postprocessed output (text) files

Using WWW browser - supercomputing in PDA/mobil



Czech Technical University in Prague

Faculty of Information Technology

Department of Software Engineering

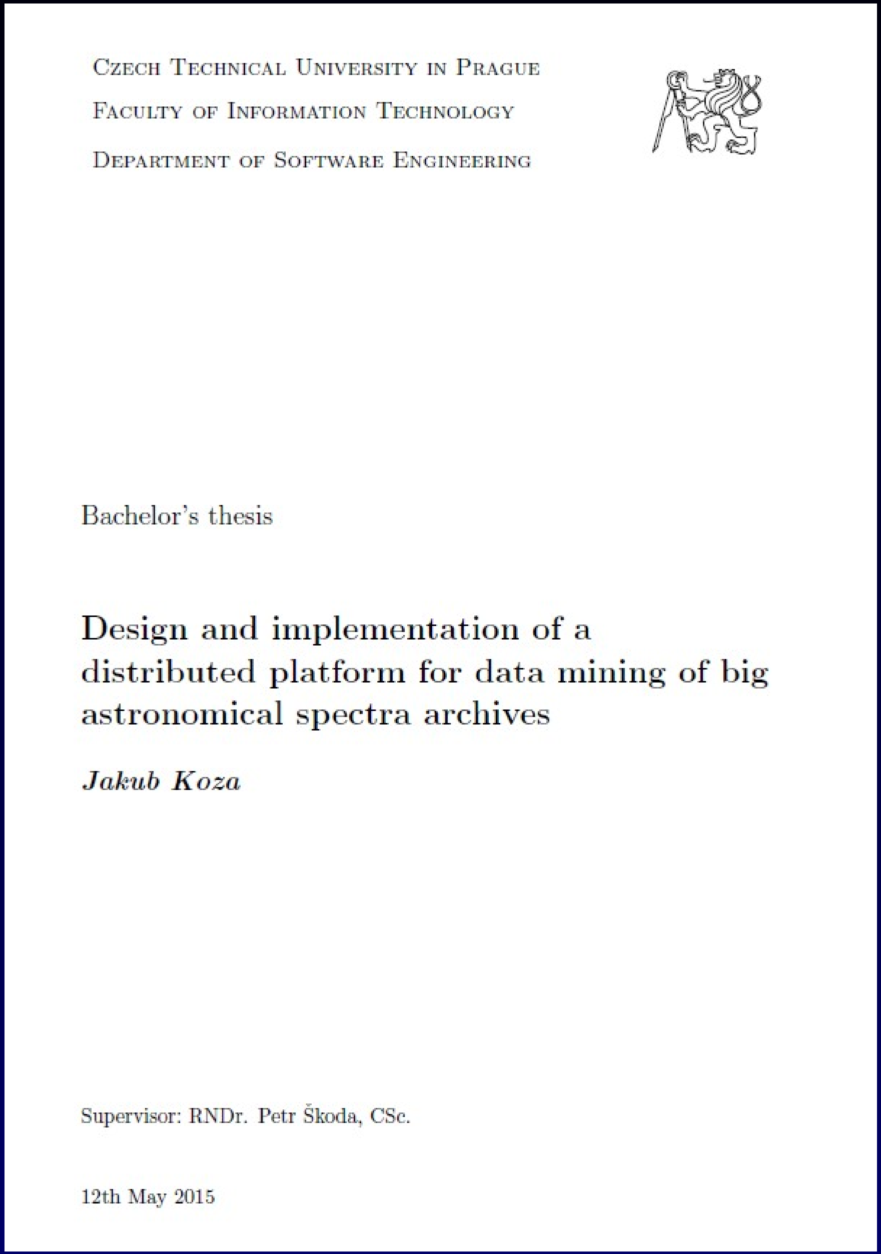
Bachelor's thesis

VO-KOREL, server for astronomical cloud computing

Lumír Mrkva

Supervisor: RNDr. Petr Škoda, CSc.

18th May 2012



Czech Technical University in Prague

Faculty of Information Technology

Department of Software Engineering

Bachelor's thesis

Design and implementation of a distributed platform for data mining of big astronomical spectra archives

Jakub Koza

Supervisor: RNDr. Petr Škoda, CSc.

12th May 2015

Czech Technical University in Prague
Faculty of Information Technology
Department of Software Engineering

Master's thesis

Interactive Cloud-Based Platform for Parallelized Machine Learning of Astronomical Big Data

Bc. Jakub Koza

Supervisor: RNDr. Petr Škoda, CSc.

9th May 2017

VO-CLOUD Architecture

Distributed engine

MASTER (frontend)

- Database of users and their experiments
- Visualization
- Scheduling
- Load balancing
- SHARED DATA STORAGE - controlled access (Big Data)

WORKERS (backend)

- Computation [+ output for visualization]

Sources of Spectra

Getting spectra + store

(restricted access – big files)

Files

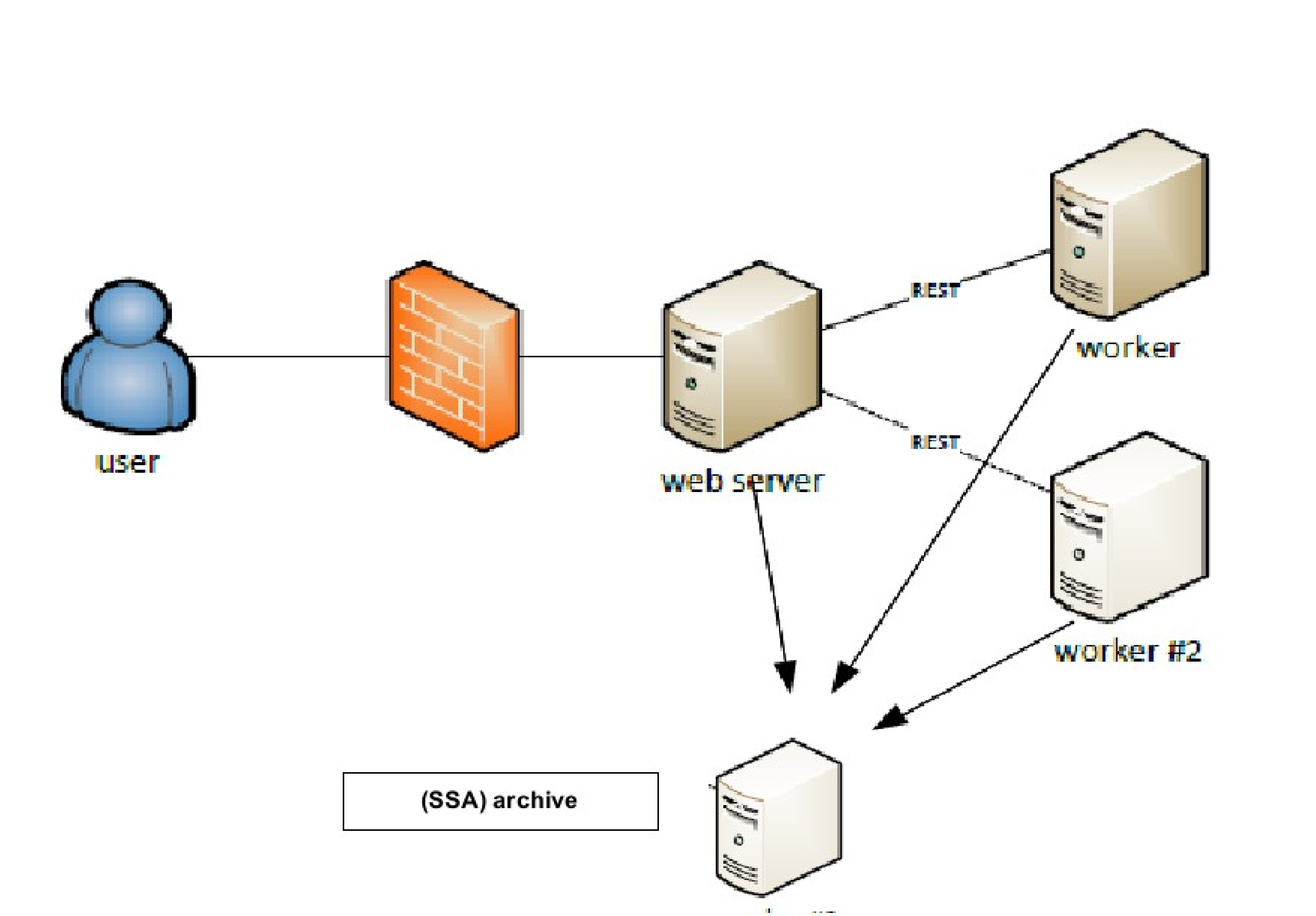
- UPLOAD from given local directory (recursive)
- DOWNLOAD by http + index, FTP (recursive)

VOTable

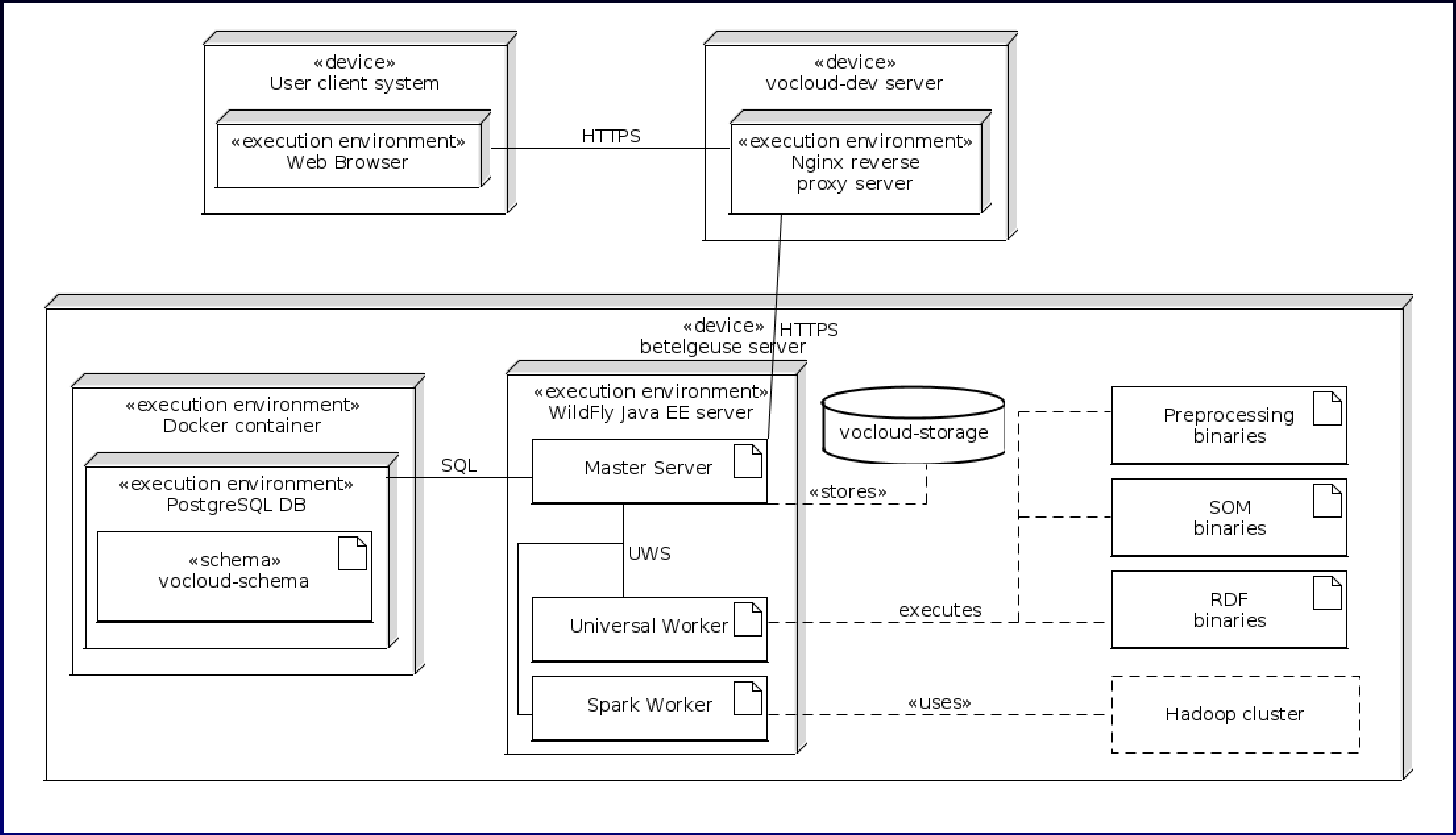
- UPLOAD VOTable (e.g. prepared in TOPCAT - meta)
- REMOTE VOTable
  - SSAP query + Accref
    - + DataLink + SODA

SAMP control - send to SPLAT

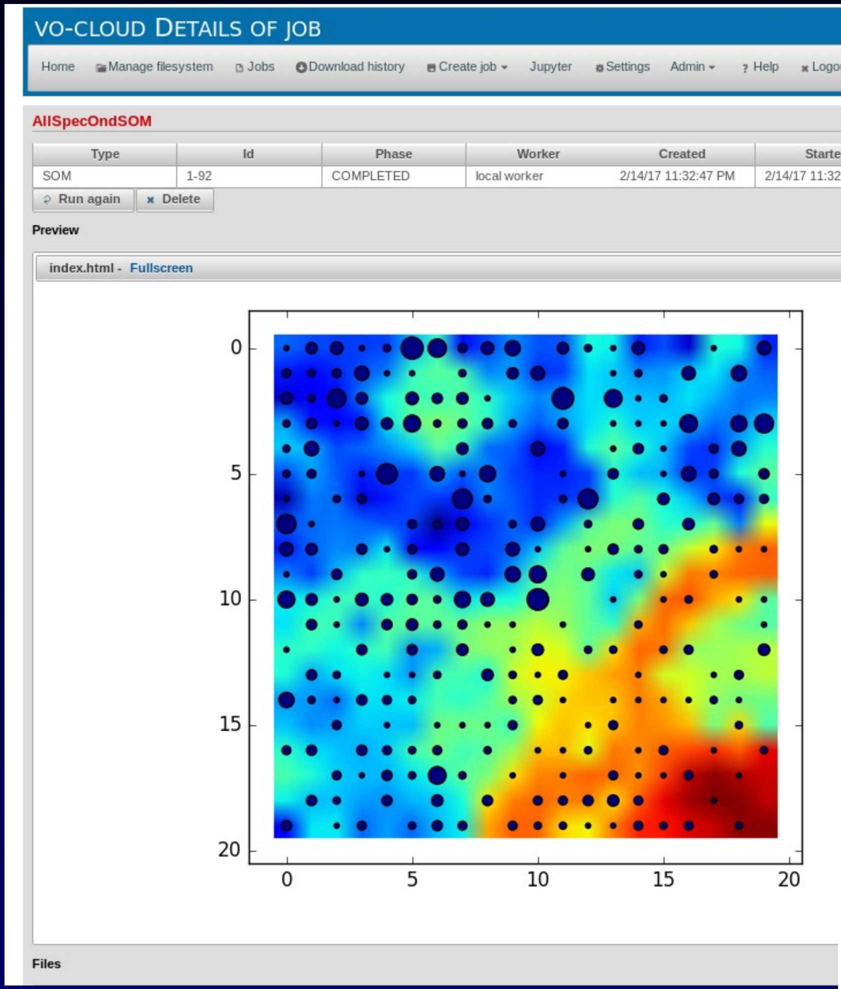
Machine Learning of BIG Archive



VO-CLOUD deployment



SOM Worker example



VO-CLOUD DETAILS OF JOB

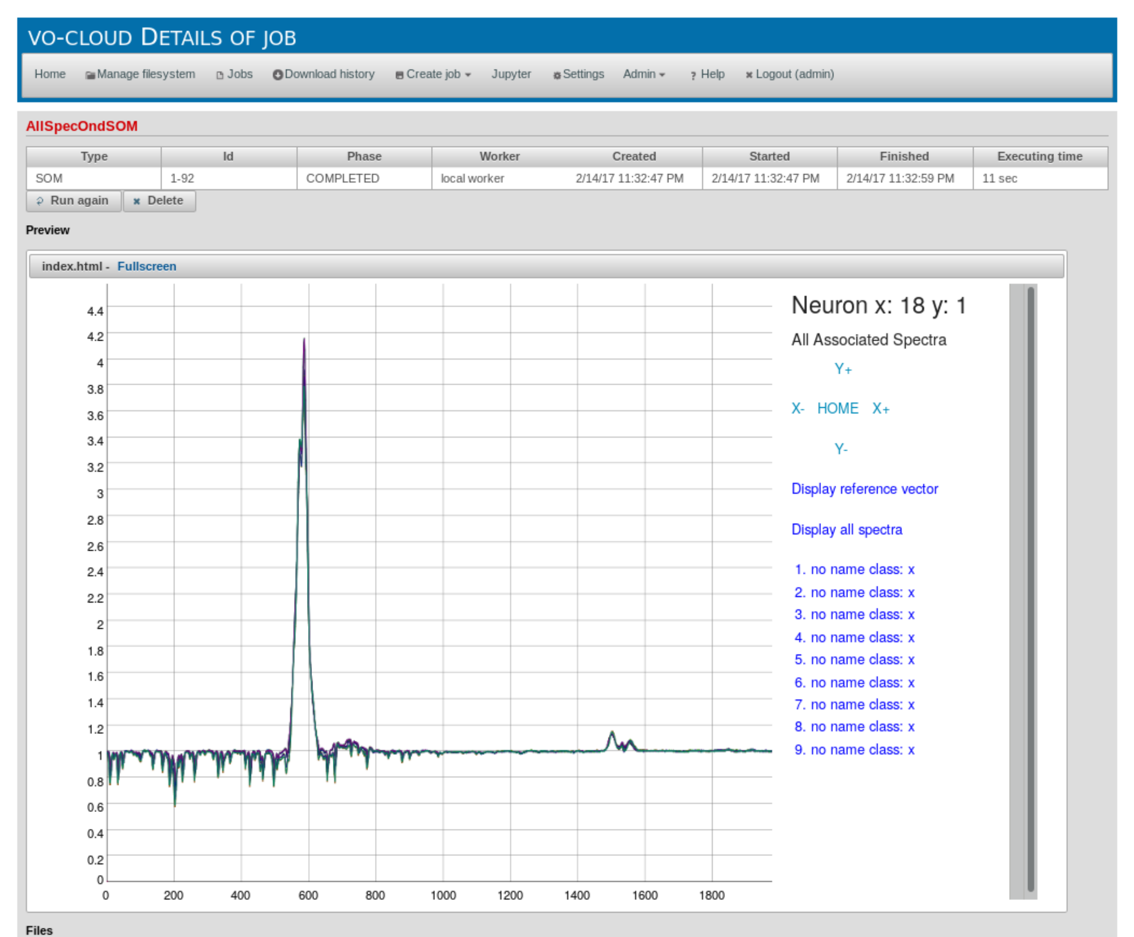
AllSpecOndSOM

Type	Id	Phase	Worker	Created	Started	Finished	Executing time
SOM	1-92	COMPLETED	local worker	2/14/17 11:32:47 PM	2/14/17 11:32:47 PM	2/14/17 11:32:59 PM	11 sec

Preview

index.html - Fullscreen

Files



VO-CLOUD DETAILS OF JOB

AllSpecOndSOM

Type	Id	Phase	Worker	Created	Started	Finished	Executing time
SOM	1-92	COMPLETED	local worker	2/14/17 11:32:47 PM	2/14/17 11:32:47 PM	2/14/17 11:32:59 PM	11 sec

Preview

index.html - Fullscreen

Neuron x: 18 y: 1

All Associated Spectra

Display reference vector

Display all spectra

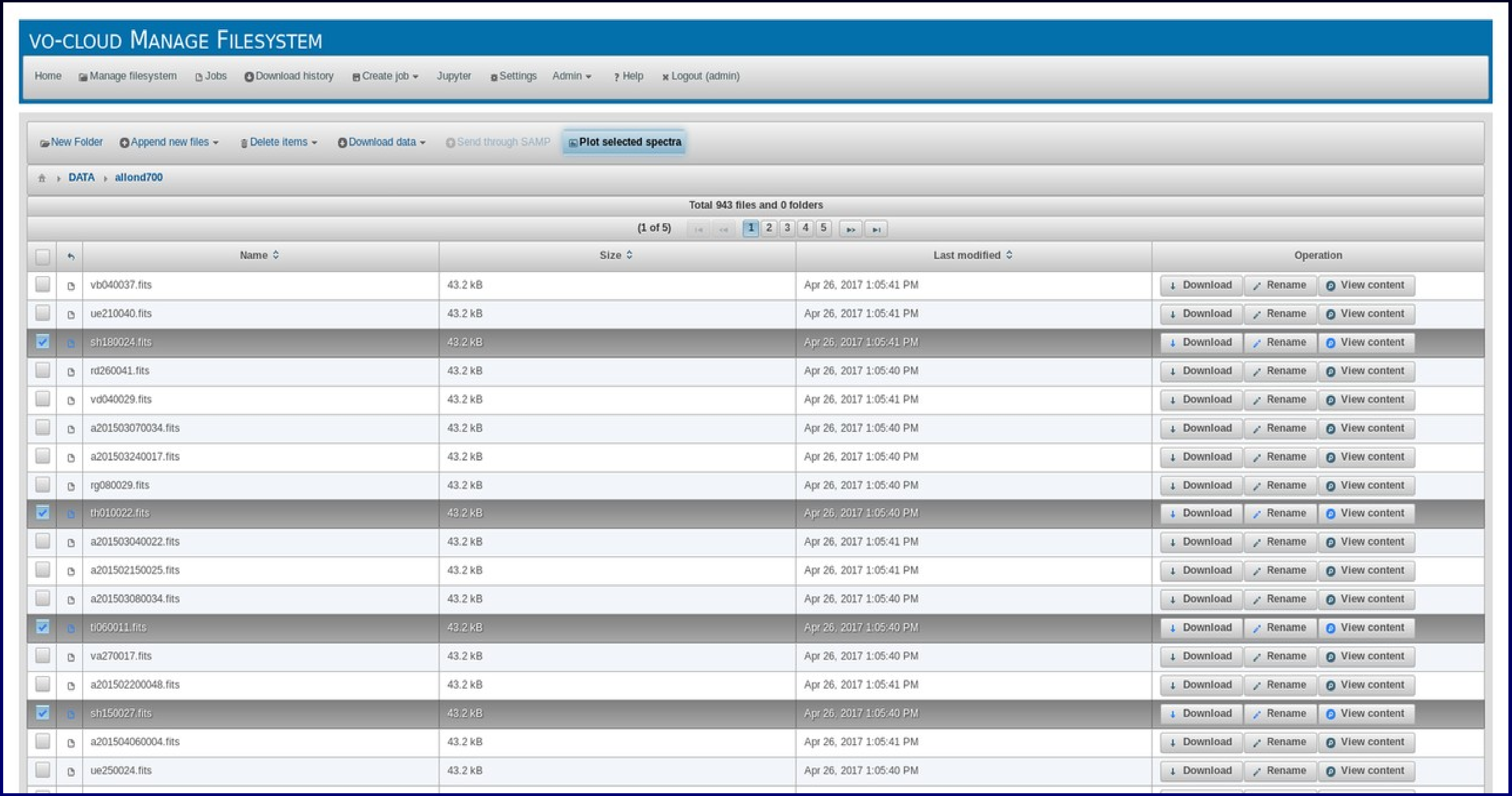
1. no name class: x
2. no name class: x
3. no name class: x
4. no name class: x
5. no name class: x
6. no name class: x
7. no name class: x
8. no name class: x
9. no name class: x

Files

VO-CLOUD spectra visualisation

- Big Data visualization (thousands spectra)
- Implemented in Python using Matplotlib
- Can visualise multiple selected spectra files
- Figure generated on server-side and then transfered to client
- User can use panning, zooming and export to different formats
- Uses WebSocket for server-client communication

VO-CLOUD spectra visualisation



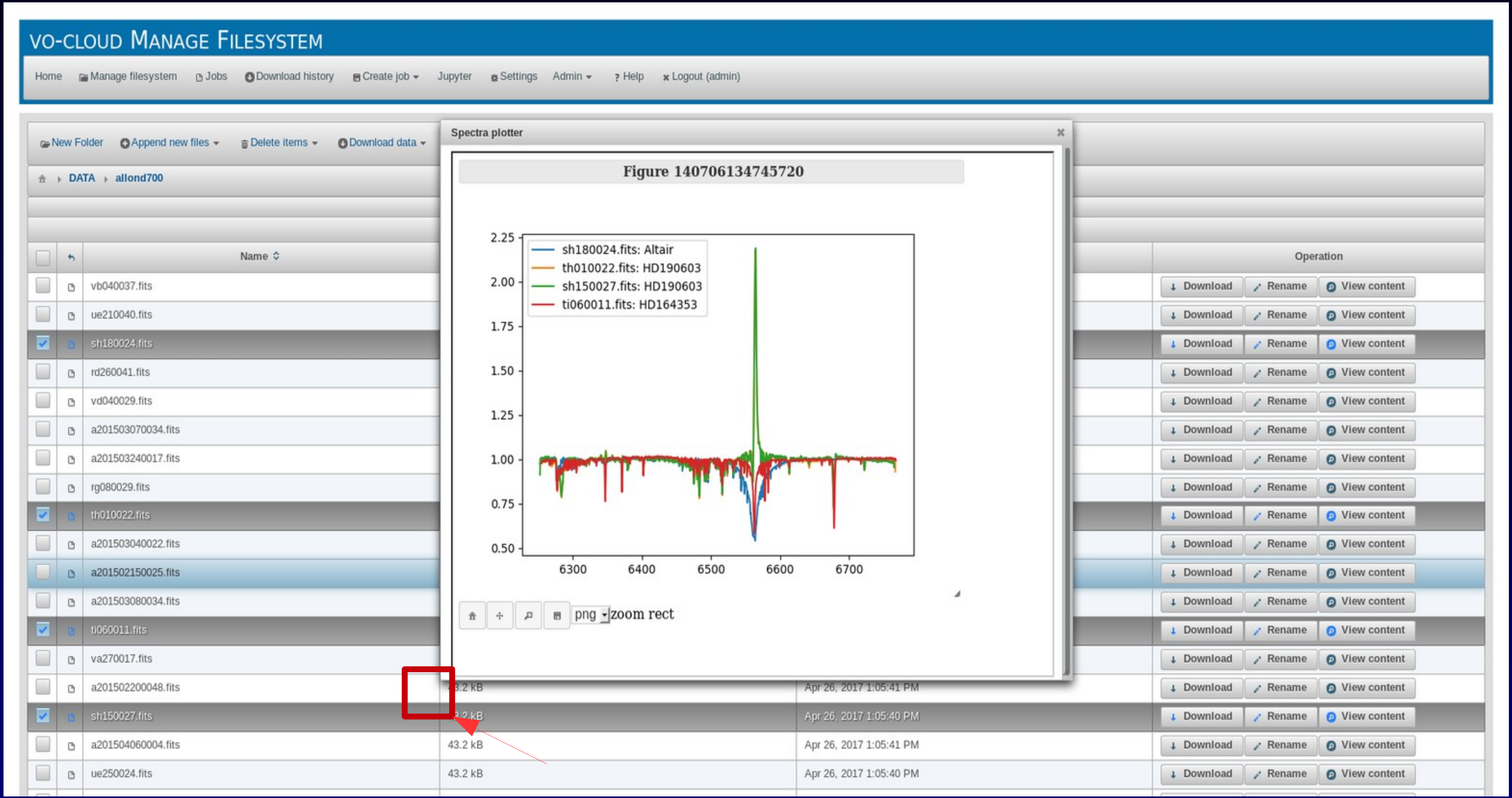
DATA › allond700

Total 943 files and 0 folders

(1 of 5) 1 2 3 4 5

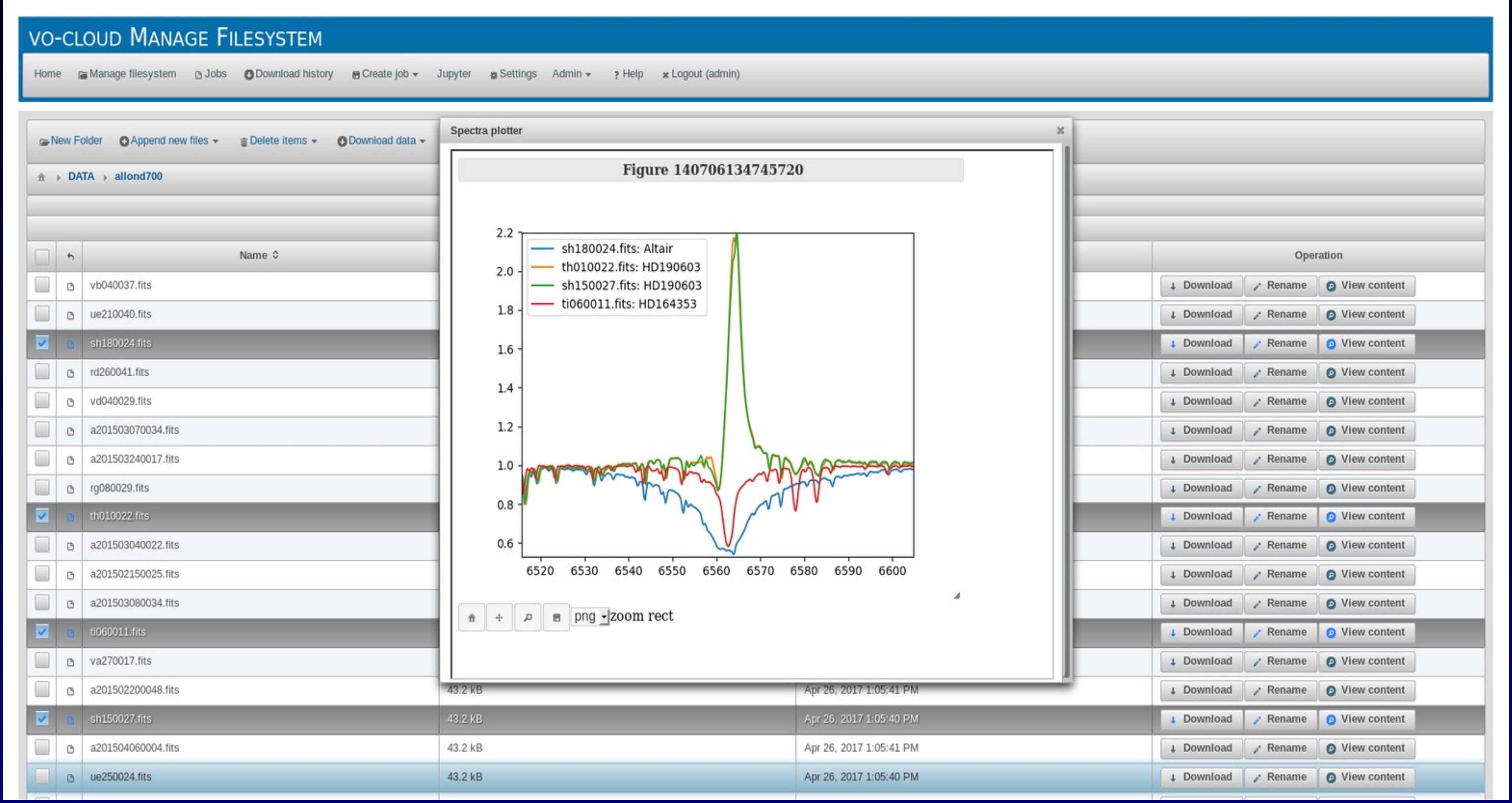
□		Name	Size	Last modified	Operation
□		vb040037.fits	43.2 kB	Apr 26, 2017 1:05:41 PM	Download Rename View content
□		ue210040.fits	43.2 kB	Apr 26, 2017 1:05:41 PM	Download Rename View content
☑		sh180024.fits	43.2 kB	Apr 26, 2017 1:05:41 PM	Download Rename View content
□		rd260041.fits	43.2 kB	Apr 26, 2017 1:05:40 PM	Download Rename View content
□		vd040029.fits	43.2 kB	Apr 26, 2017 1:05:41 PM	Download Rename View content
□		a201503070034.fits	43.2 kB	Apr 26, 2017 1:05:40 PM	Download Rename View content
□		a201503240017.fits	43.2 kB	Apr 26, 2017 1:05:40 PM	Download Rename View content
□		rg080029.fits	43.2 kB	Apr 26, 2017 1:05:40 PM	Download Rename View content
☑		th010022.fits	43.2 kB	Apr 26, 2017 1:05:40 PM	Download Rename View content
□		a201503040022.fits	43.2 kB	Apr 26, 2017 1:05:41 PM	Download Rename View content
□		a201502150025.fits	43.2 kB	Apr 26, 2017 1:05:41 PM	Download Rename View content
□		a201503080034.fits	43.2 kB	Apr 26, 2017 1:05:40 PM	Download Rename View content
☑		ti060011.fits	43.2 kB	Apr 26, 2017 1:05:40 PM	Download Rename View content
□		va270017.fits	43.2 kB	Apr 26, 2017 1:05:40 PM	Download Rename View content
□		a201502200048.fits	43.2 kB	Apr 26, 2017 1:05:41 PM	Download Rename View content
☑		sh150027.fits	43.2 kB	Apr 26, 2017 1:05:40 PM	Download Rename View content
□		a201504060004.fits	43.2 kB	Apr 26, 2017 1:05:41 PM	Download Rename View content
□		ue250024.fits	43.2 kB	Apr 26, 2017 1:05:40 PM	Download Rename View content

VO-CLOUD spectra visualisation



□		Name ⇕			Operation
□		vb040037.fits			Download Rename View content
□		ue210040.fits			Download Rename View content
☑		sh180024.fits			Download Rename View content
□		rd260041.fits			Download Rename View content
□		vd040029.fits			Download Rename View content
□		a201503070034.fits			Download Rename View content
□		a201503240017.fits			Download Rename View content
□		rg080029.fits			Download Rename View content
☑		th010022.fits			Download Rename View content
□		a201503040022.fits			Download Rename View content
□		a201502150025.fits			Download Rename View content
□		a201503080034.fits			Download Rename View content
☑		ti060011.fits			Download Rename View content
□		va270017.fits			Download Rename View content
□		a201502200048.fits	3.2 kB	Apr 26, 2017 1:05:41 PM	Download Rename View content
☑		sh150027.fits	3.2 kB	Apr 26, 2017 1:05:40 PM	Download Rename View content
□		a201504060004.fits	43.2 kB	Apr 26, 2017 1:05:41 PM	Download Rename View content
□		ue250024.fits	43.2 kB	Apr 26, 2017 1:05:40 PM	Download Rename View content

VO-CLOUD spectra visualisation

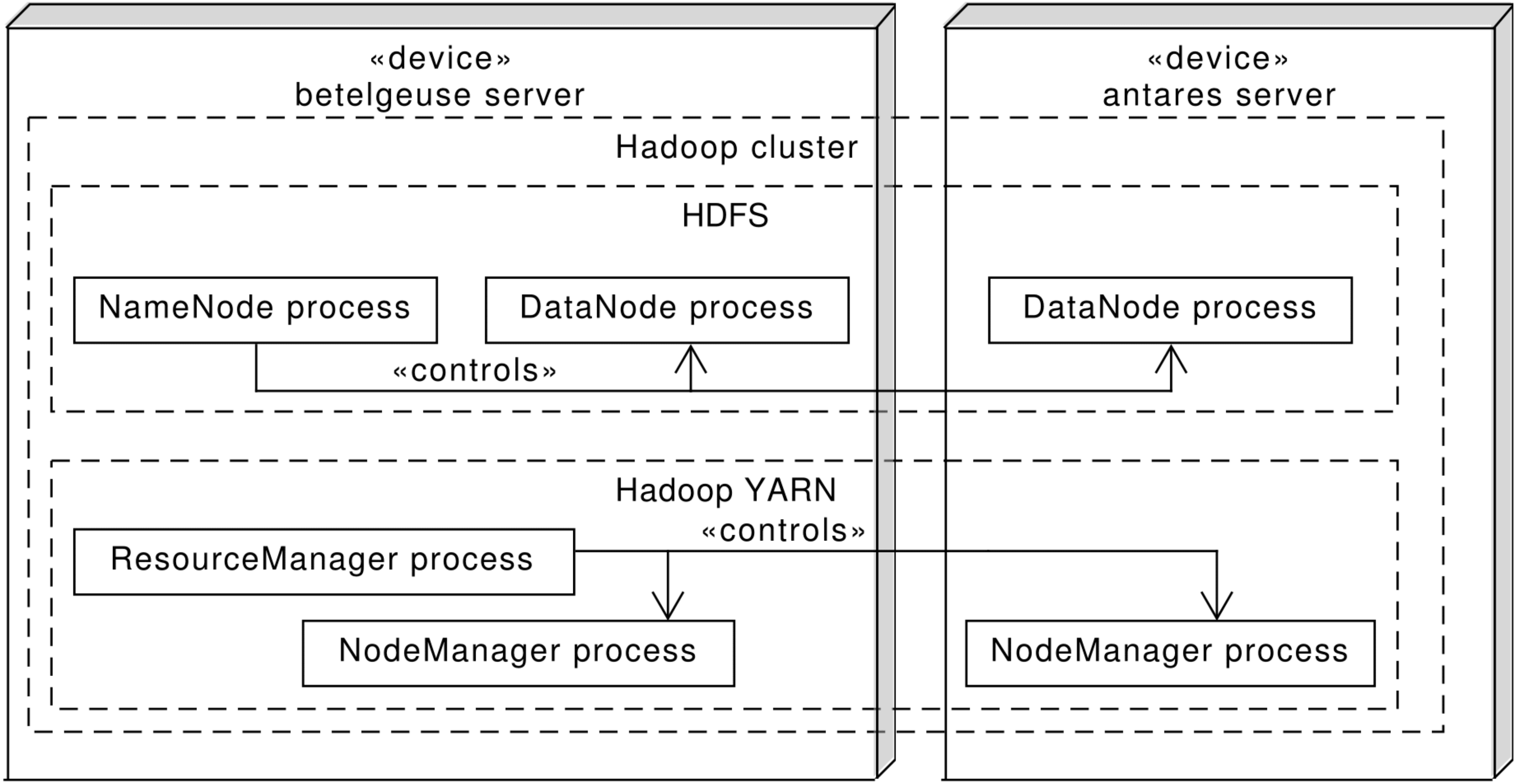


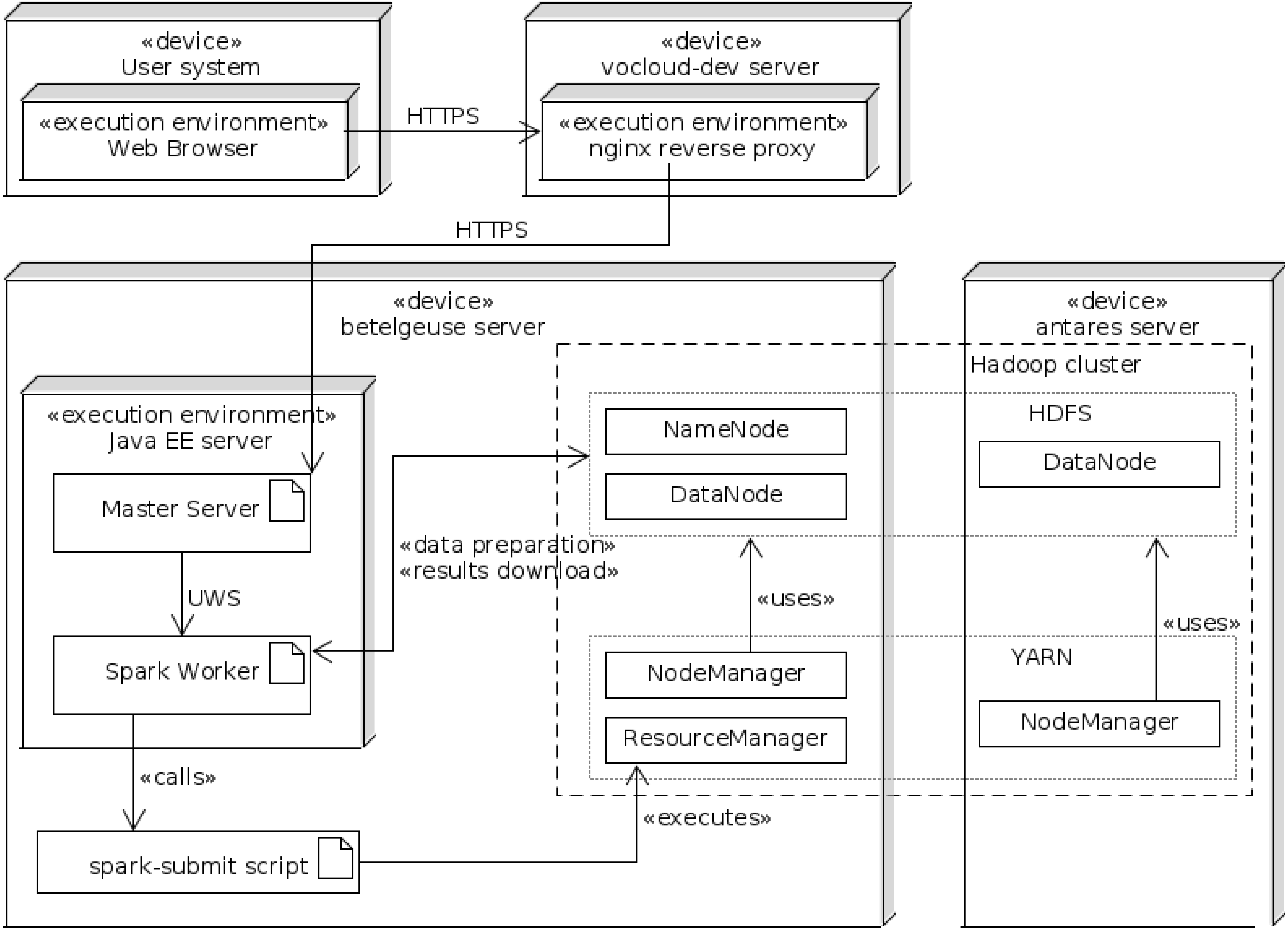
☐		Name			Operation
☐		vb040037.fits			Download Rename View content
☐		ue210040.fits			Download Rename View content
☑		sh180024.fits			Download Rename View content
☐		rd260041.fits			Download Rename View content
☐		vd040029.fits			Download Rename View content
☐		a201503070034.fits			Download Rename View content
☐		a201503240017.fits			Download Rename View content
☐		rg080029.fits			Download Rename View content
☑		th010022.fits			Download Rename View content
☐		a201503040022.fits			Download Rename View content
☐		a201502150025.fits			Download Rename View content
☐		a201503080034.fits			Download Rename View content
☑		ti060011.fits			Download Rename View content
☐		va270017.fits			Download Rename View content
☐		a201502200048.fits	43.2 kB	Apr 26, 2017 1:05:41 PM	Download Rename View content
☑		sh150027.fits	43.2 kB	Apr 26, 2017 1:05:40 PM	Download Rename View content
☐		a201504060004.fits	43.2 kB	Apr 26, 2017 1:05:41 PM	Download Rename View content
☐		ue250024.fits	43.2 kB	Apr 26, 2017 1:05:40 PM	Download Rename View content

Hadoop cluster infrastructure

- HDFS – Distributed filesystem utilized as a storage in Hadoop/Spark jobs
  - NameNode – One process per cluster. Contains information about all files saved inside HDFS
  - DataNode – One process per each node in cluster. Stores individual file data blocks.
- YARN – Resource manager and scheduler for Hadoop/Spark jobs.
  - ResourceManager – Main process managing resources and scheduling jobs. One process per cluster.
  - NodeManager – Process executing assigned work on each node. One process per each cluster node.
  - Problem of millions of small files (FITS) – Apache AVRO (Sequence files)

Hadoop cluster deployment





«device»
User system

«execution environment»
Web Browser

HTTPS

«device»
vocloud-dev server

«execution environment»
nginx reverse proxy

HTTPS

«device»
betelgeuse server

«execution environment»
Java EE server

Master Server

UWS

Spark Worker

«calls»

spark-submit script

«data preparation»
«results download»

«executes»

«device»
antares server

Hadoop cluster

HDFS

NameNode

DataNode

DataNode

«uses»

«uses»

YARN

NodeManager

ResourceManager

NodeManager

Spark Worker in VO-CLOUD

- Accepts JSON configuration
- Downloads requested files from the Master server to HDFS
- Executes spark-submit script using implicit parameters or parameters present in JSON configuration
- Awaits spark-submit script completion
- Downloads requested files from the HDFS

- Master FILESYSTEM – NFS emulation for HDFS

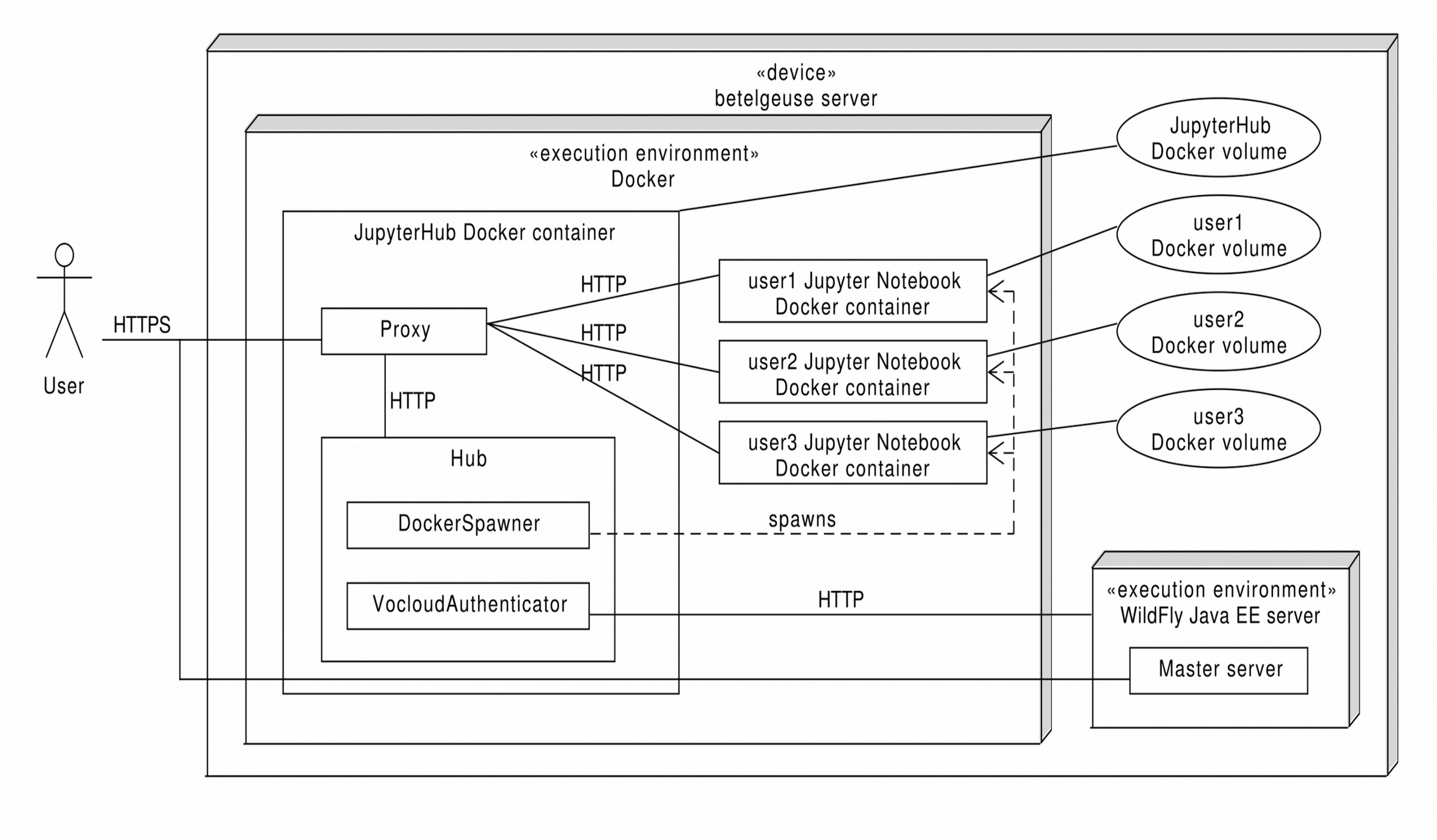
JupyterHub

- Consists of Proxy, Hub and individual Jupyter Notebook server instances
- One Jupyter Notebook server instance per authenticated user
- JupyterHub runs as Docker container
- JupyterHub spawns additional Docker containers – one for each Jupyter Notebook server instance
- Users are isolated from each other and from hosting system itself

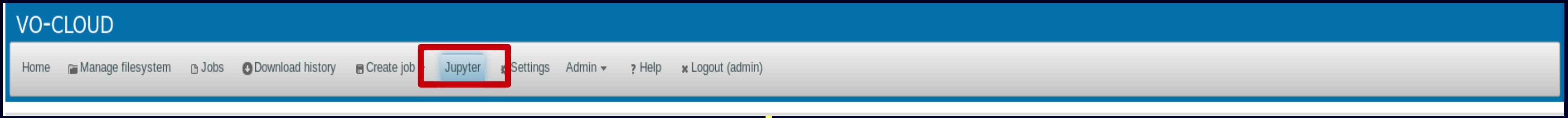
JupyterHub authentication

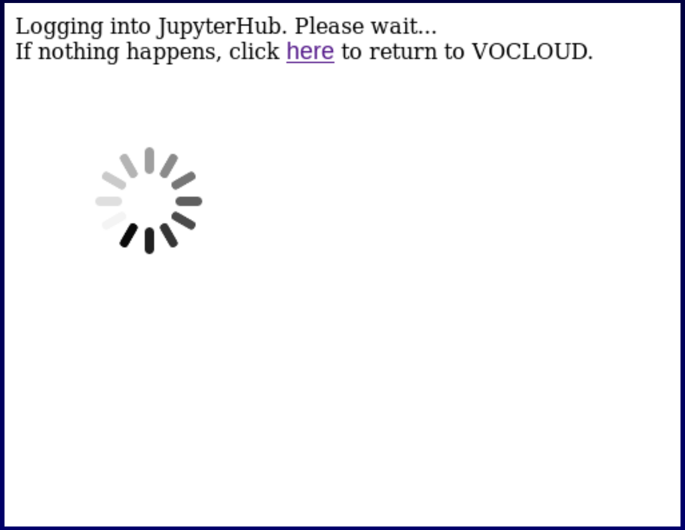
1) VO-CLOUD generates temporal token and passes it to user‘s web browser
2) Web browser uses username and token for JupyterHub authentication
3) JupyterHub uses vocloud-authenticator package for authentication
4) Authenticator asks VO-CLOUD whether token is valid for relevant username
5) JupyterHub spawns new Docker container with the new Jupyter Notebook server instance for the user

JupyterHub deployment

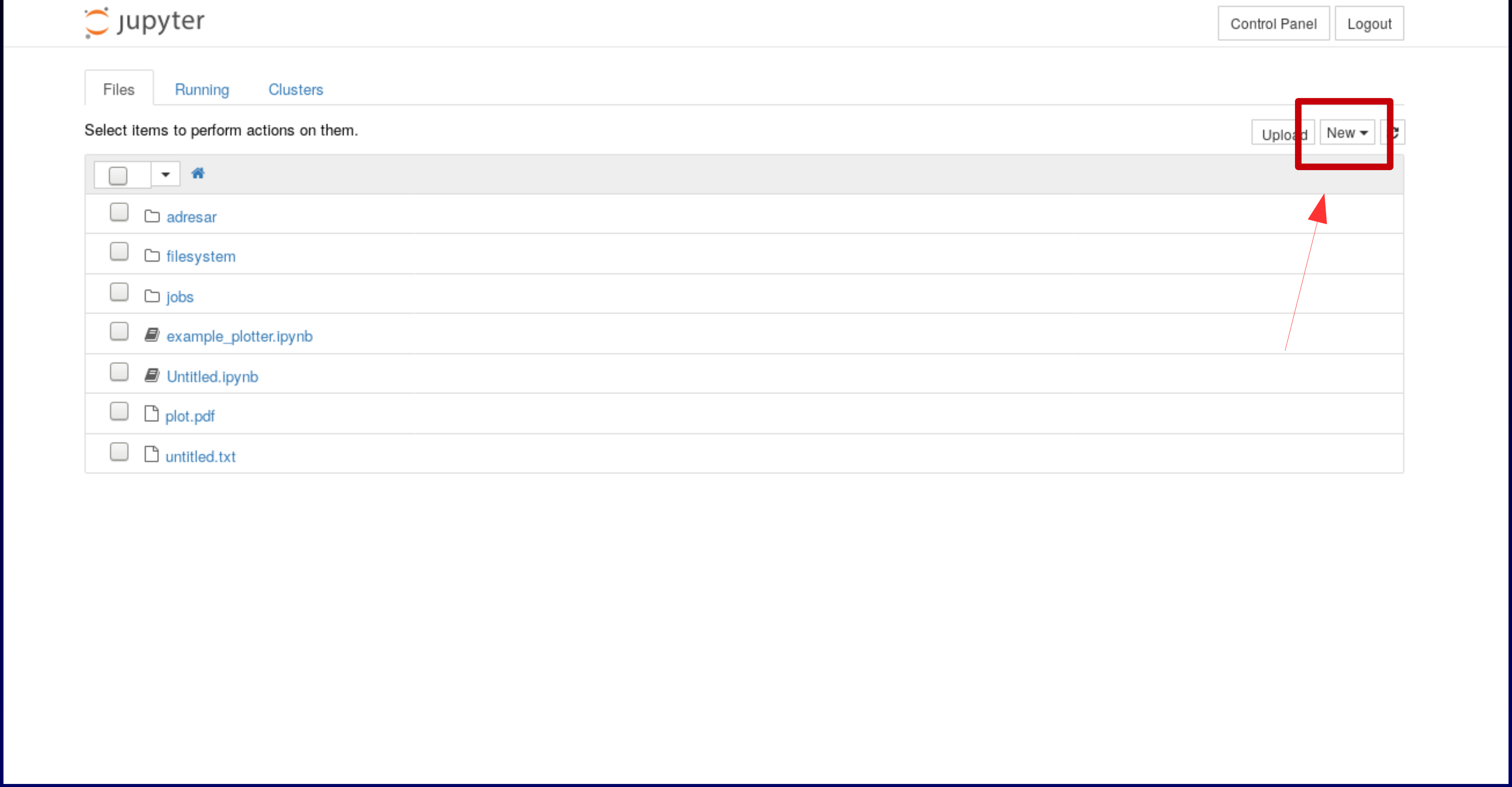


JupyterHub example




Logging into JupyterHub. Please wait...
If nothing happens, click here to return to VOCLOUD.

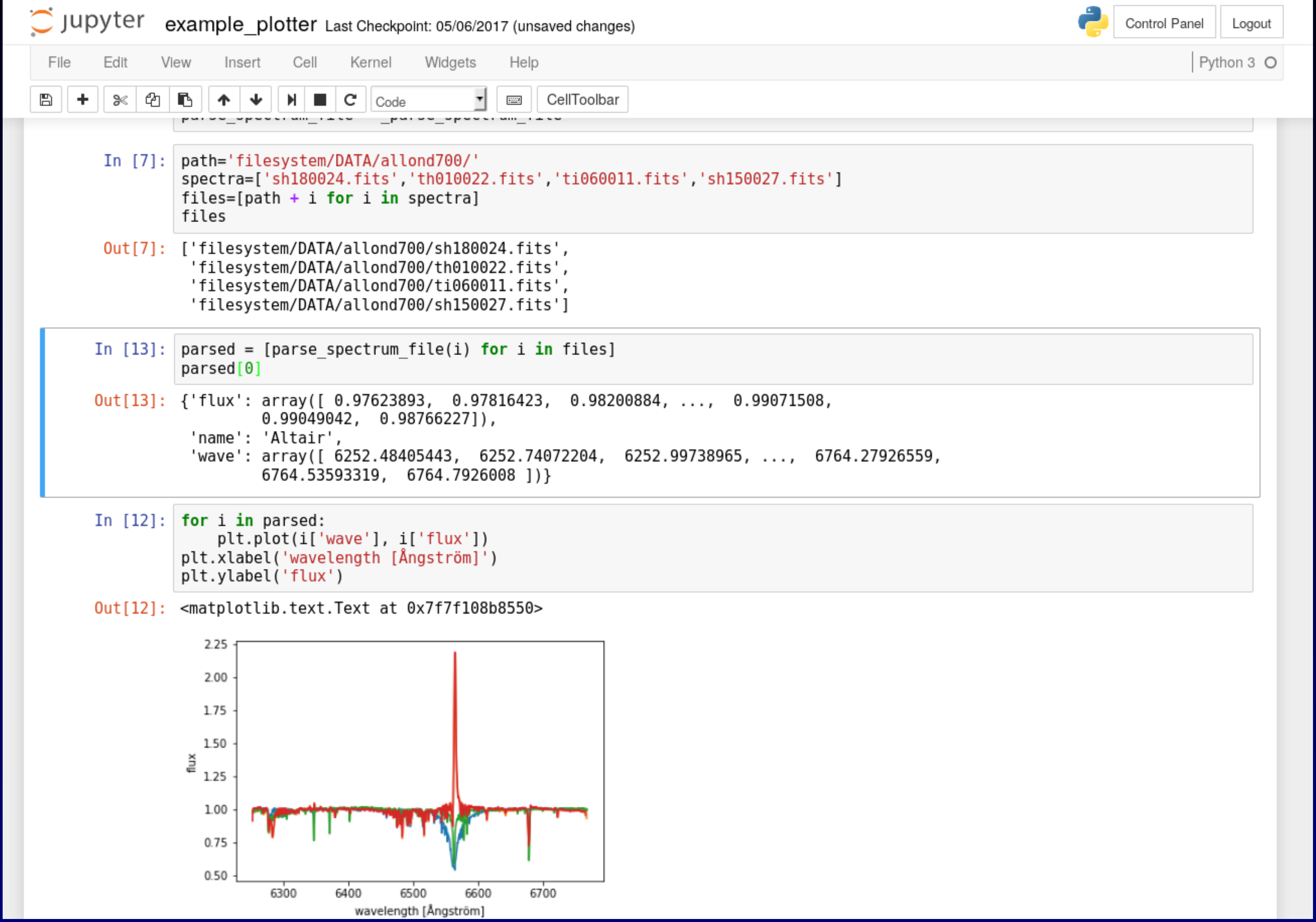
JupyterHub example



Select items to perform actions on them.

- □ adresar
- □ filesystem
- □ jobs
- □ example_plotter.ipynb
- □ Untitled.ipynb
- □ plot.pdf
- □ untitled.txt

JupyterHub example



In [7]:
```
path='filesystem/DATA/allond700/'
spectra=['sh180024.fits','th010022.fits','ti060011.fits','sh150027.fits']
files=[path + i for i in spectra]
files
```

Out[7]:
```
['filesystem/DATA/allond700/sh180024.fits',
 'filesystem/DATA/allond700/th010022.fits',
 'filesystem/DATA/allond700/ti060011.fits',
 'filesystem/DATA/allond700/sh150027.fits']
```

In [13]:
```
parsed = [parse_spectrum_file(i) for i in files]
parsed[0]
```

Out[13]:
```
{'flux': array([ 0.97623893,  0.97816423,  0.98200884, ...,  0.99071508,
         0.99049042,  0.98766227]),
 'name': 'Altair',
 'wave': array([ 6252.48405443,  6252.74072204,  6252.99738965, ...,  6764.27926559,
         6764.53593319,  6764.7926008 ])}
```

In [12]:
```
for i in parsed:
    plt.plot(i['wave'], i['flux'])
plt.xlabel('wavelength [Ångström]')
plt.ylabel('flux')
```

Out[12]:
```
<matplotlib.text.Text at 0x7f7f108b8550>
```

Conclusions

- VO-CLOUD is now very powerful machine learning environment capable of visualization of Big data
- It can spawn jobs on remote Spark cluster
- Provides sandbox for playing with big data in Jupyter notebook ON BIG SERVER (memory, CPU, GPU)

  but
- Still missing important capabiliites
  - AVRO not part of Spark-worker
  - Using Docker but so far not deployable as a docker (compose)
  - Combines Java EE + Python (+ Scala)
- Focused on Machine learning of 1D vectors (spectra, time series) employing VO technology and protocols

Source Code

https://github.com/vodev/vocloud