VO-CLOUD for Machine Learning

Petr Škoda

Astronomical Institute Academy of Sciences Ondřejov
Czech Republic

Lukáš Lopatovský, Andrej Palička, Lumír Mrkva
Faculty of Informatics, Czech Technical University, Prague

Supported by grant GAČR 13-08195S

IVOA Interoperability meeting , Apps session 3
Banff, Alberta, Canada 12th October 2014

Concept of scientific „CLOUD"

ITERATIVE REPEATING of SAME computation (workflow)

Global non-linear optimization (Korel)

Synthetic spectra (various elements, wavelength-ranges)

Machine Learning (almost all methods)

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

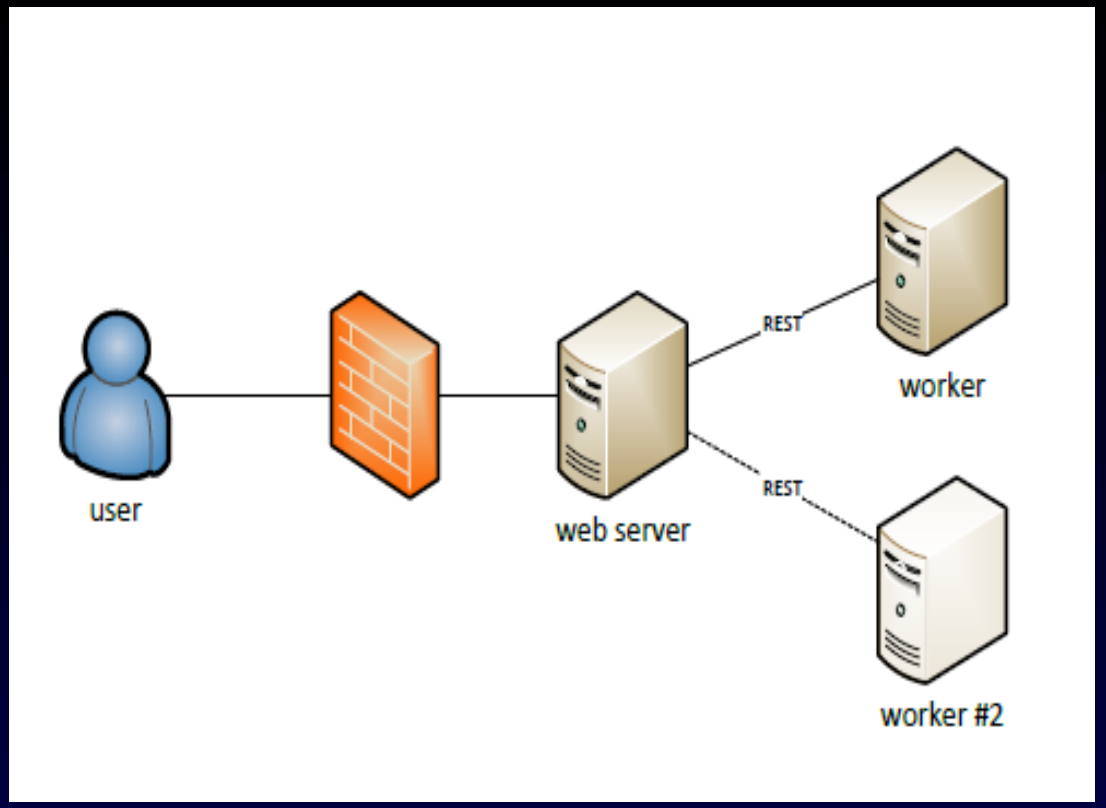
Bachelor's thesis

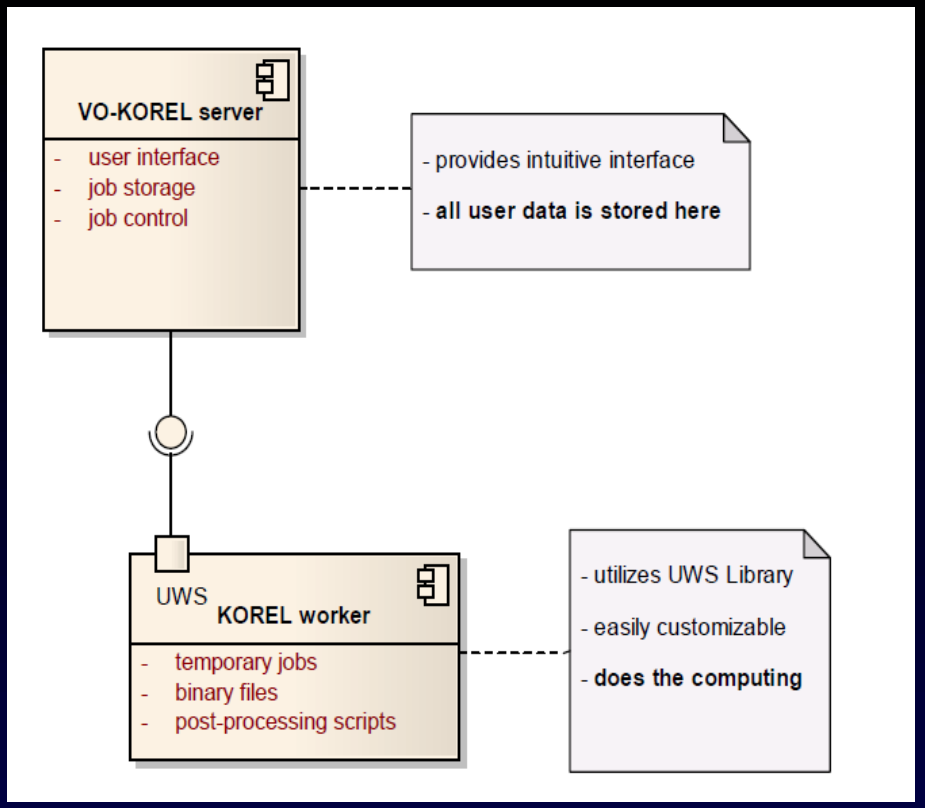
VO-KOREL, server for astronomical cloud computing

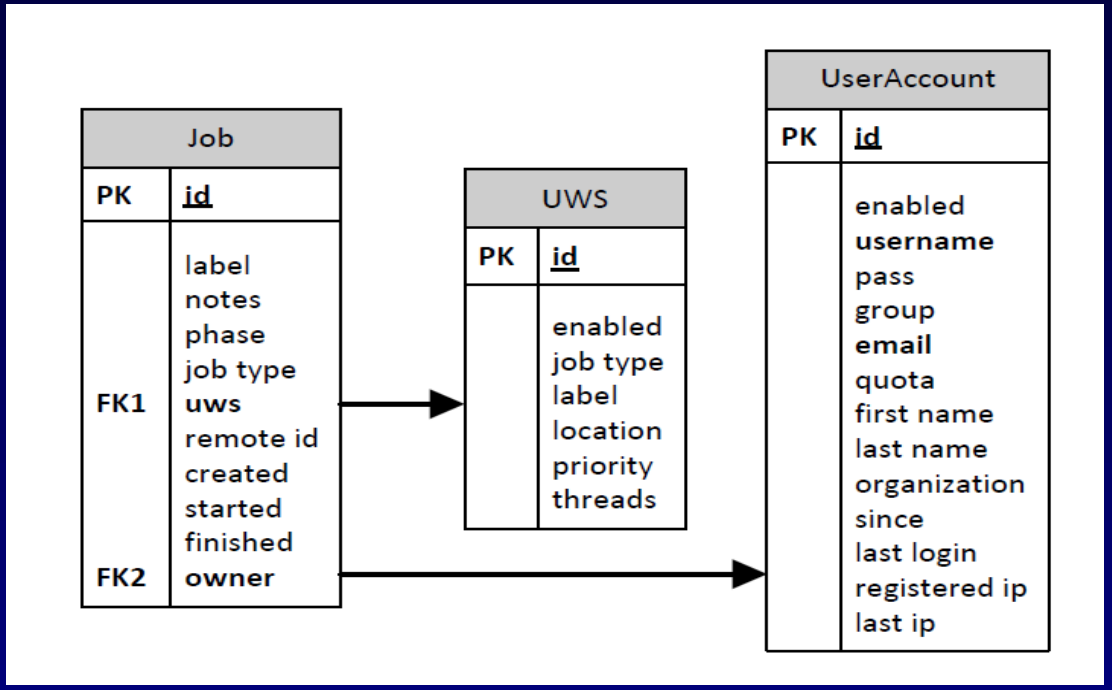
Lumír Mrkva

Supervisor: RNDr. Petr Škoda, CSc.

18th May 2012


user
web server
REST
REST
worker
worker #2


VO-KOREL server
- user interface
- job storage
- job control
- provides intuitive interface
- all user data is stored here
UWS
KOREL worker
- temporary jobs
- binary files
- post-processing scripts
- utilizes UWS Library
- easily customizable
- does the computing


Job
PK id
label
notes
phase
job type
FK1 uws
remote id
created
started
finished
FK2 owner
UWS
PK id
enabled
job type
label
location
priority
threads
UserAccount
PK id
enabled
username
pass
group
email
quota
first name
last name
organization
since
last login
registered ip
last ip

CREATE job

server selects the less utilitized worker of same type (KOREL,SOM, RF)

worker must contain binary+LIBRARIES of computing engine

server creates job on worker – gives link to *params.zip* stored on server (here are all user inputs = data +params + control info)

worker stores the link (knows how to download)

server stores the jobid obtained from worker

job is in PHASE=PENDING

RUN job

User wants to run the job (Save and run or Run again)

Server send PHASE=RUN to the worker

Worker downolads *params.zip* using the stored link

Server starts polling the PHASE of job on worker

If PHASE=COMPLETED, ABORTED, ERROR

Server downloads the results from worker (encrypted)

Server sends ACTION=DELETE to the worker – destroys job

ABORT job

User decides to ABORT the job

Server send PHASE=ABORT to the worker

Worker stops the computation (signal to <PROGRAM> binary)

Server downloads the results from worker

Server sends ACTION=DELETE to the worker – destroys job

SW Requirements

SERVER

PostgreSQL (user database)

Storage space for user experiments (quota)

Glassfish 3.1 or Jboss (needs application server for EJB)

Visualization scripts (convert to png, test presence of data)

WORKER

Binary of computing engine of given type (KOREL, SOM)

JAVA7 + Servlet engine (Tomcat7, Jboss, GlassFish)

Advantages of this solution

Standard UWS library – no modifications

ABORT returns results !

Users isolation (user sees only his UWS jobs on server)

Security (hash id – direct connection to worker knows only server, encryption of transfered data possible)

Scalability (more workers added dynamically – admin UI)

Workers for different computations may be prepared (load)

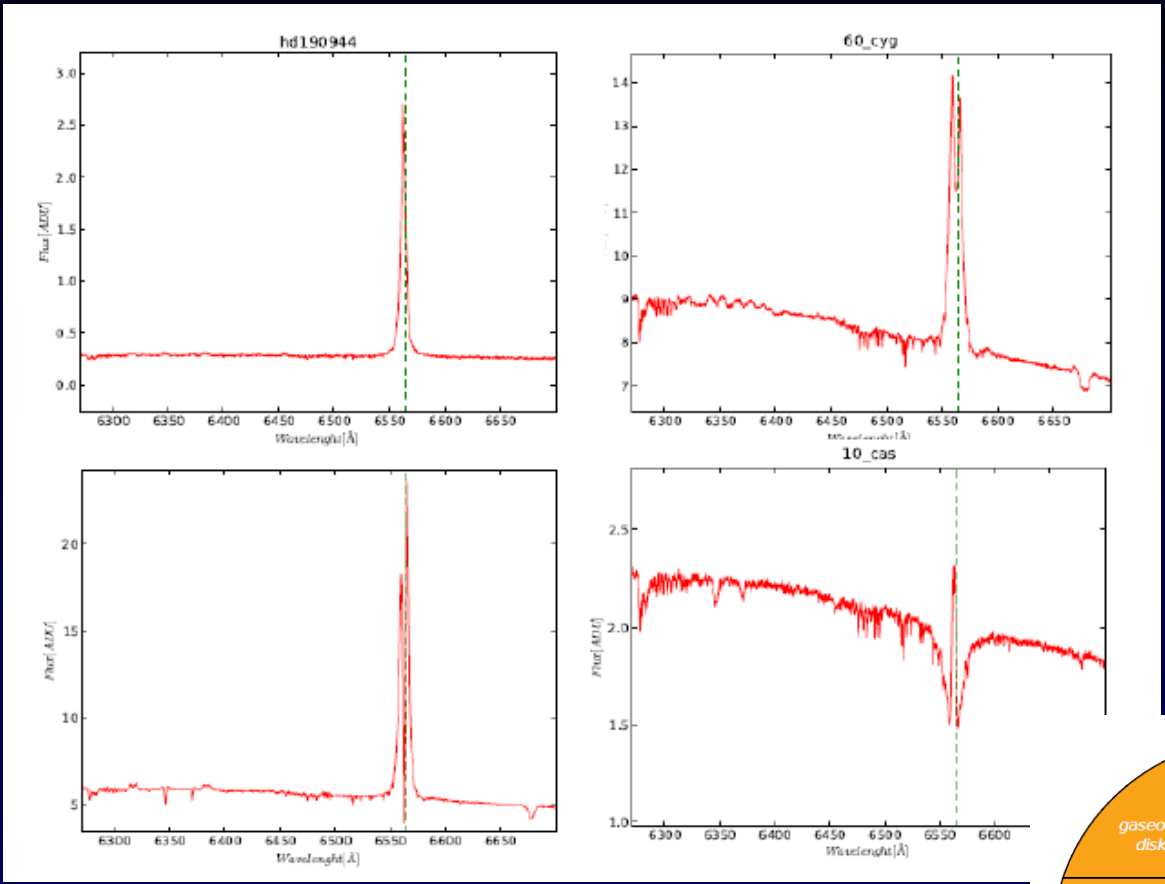
User friendly GUI on server – browser on phones (AJAX)

Can be interfaced to Grid or cluster queuing systems

worker sends the source of computing engine + data

Machine Learning of Spectra

Use case: ML of spectra profile of Halpha line (Be stars)

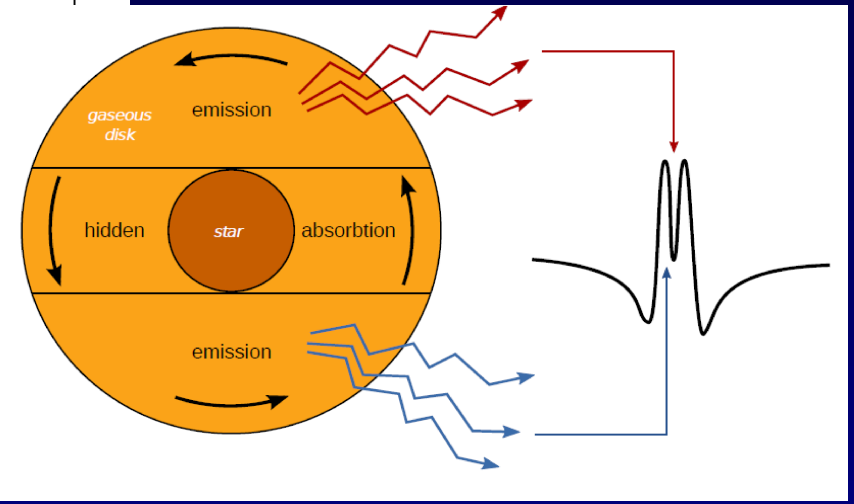


Be stars

Disk or envelope

Rotates, Hot

Origin ?????



Machine Learning of Spectra
Science case

Ondřejov 2m Perek Telescope – 1700/10 000 spectra

„large data“ – whole spectra

PRE-PROCESSING – Normalization, CUTOUT (SSAP+DL)

Rebinning (same wavelegth points)

Reduction of dimensionality (wavelets, LLE, PPS, ?PCA?)

Produces FEATURE VECTORS – same length, dimensions

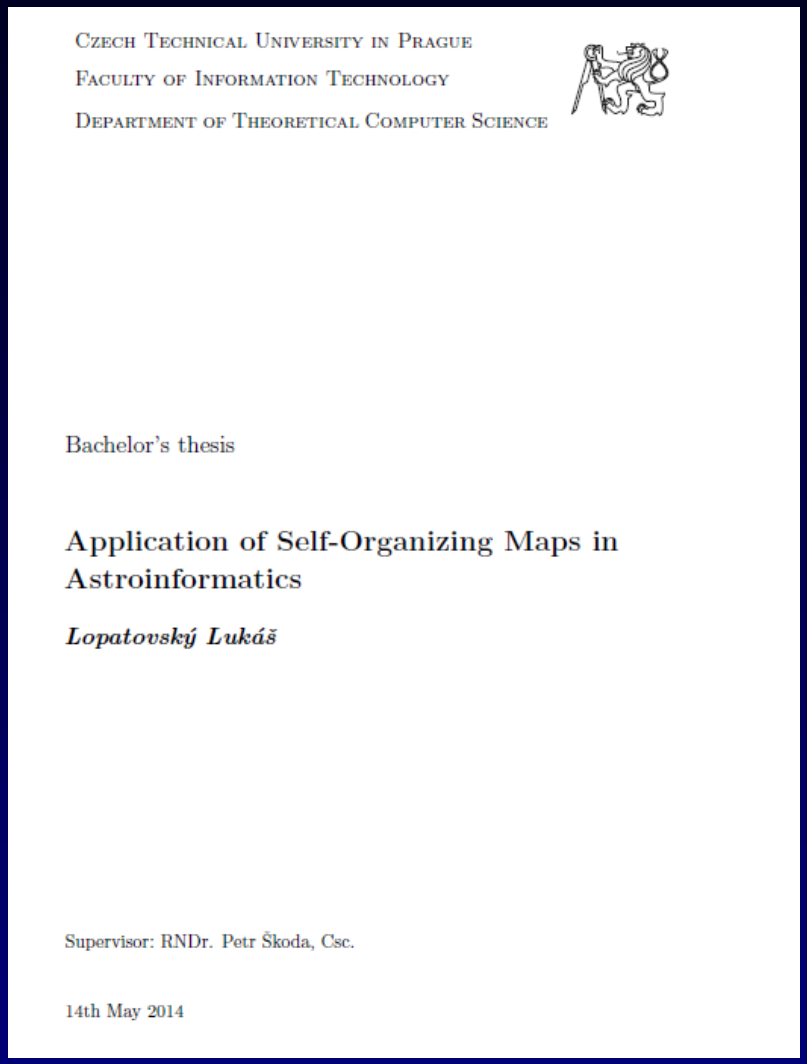
Unified wrapper running multiple applications - same call

Name-of-wrapper + parameters (json) – method as param

Gnuplot for visualization

Bachelor's Theses
Faculty of Informatics CTU Prague

Unsupervised – SOM
All data - Outliers



Czech Technical University in Prague
Faculty of Information Technology
Department of Theoretical Computer Science

Bachelor's thesis

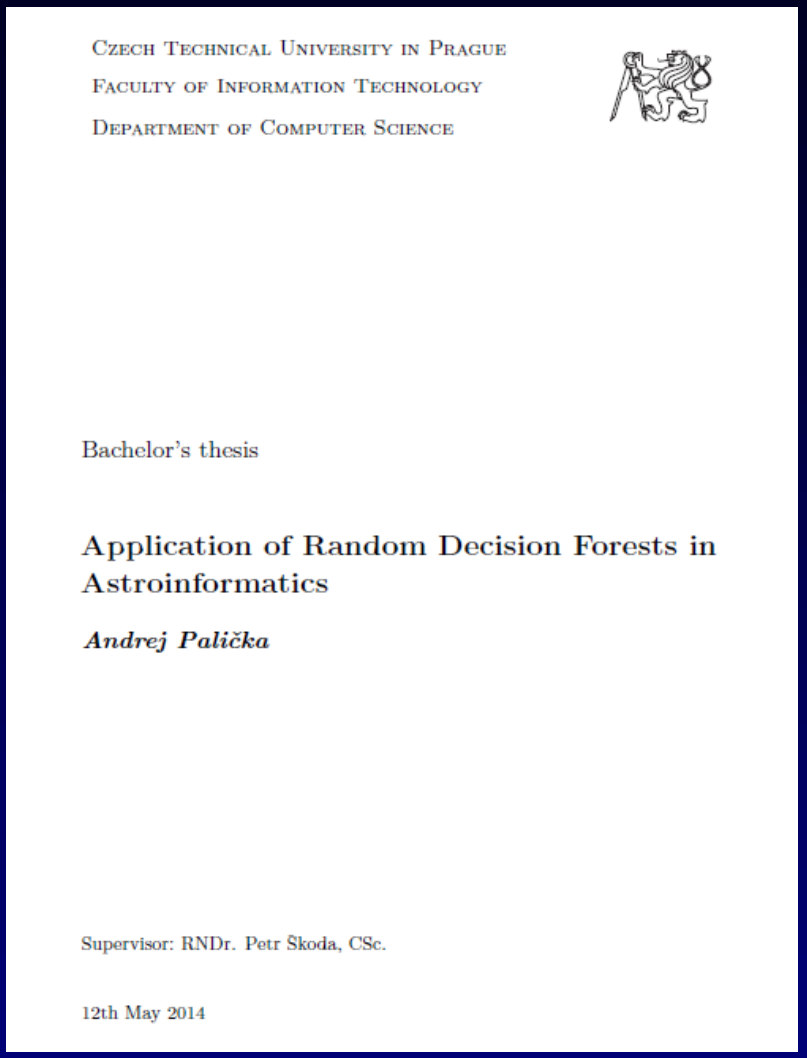
Application of Self-Organizing Maps in Astroinformatics

Lopatovský Lukáš

Supervisor: RNDr. Petr Škoda, Csc.

14th May 2014

Supervised – Classes by eye
Training+testing set



Czech Technical University in Prague
Faculty of Information Technology
Department of Computer Science

Bachelor's thesis

Application of Random Decision Forests in Astroinformatics

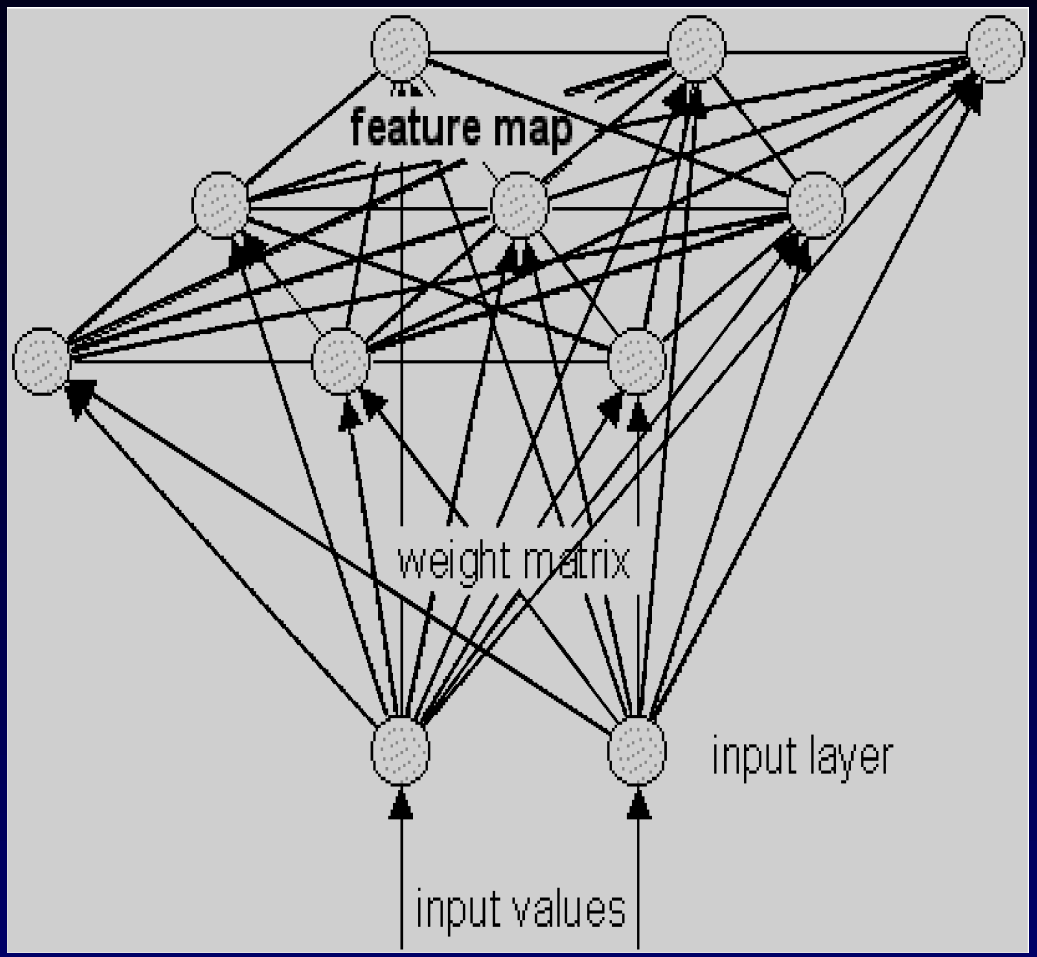
Andrej Palička

Supervisor: RNDr. Petr Škoda, CSc.

12th May 2014

Principles of SOM

Self-Organizing = Kohonen map



Association (activation) map

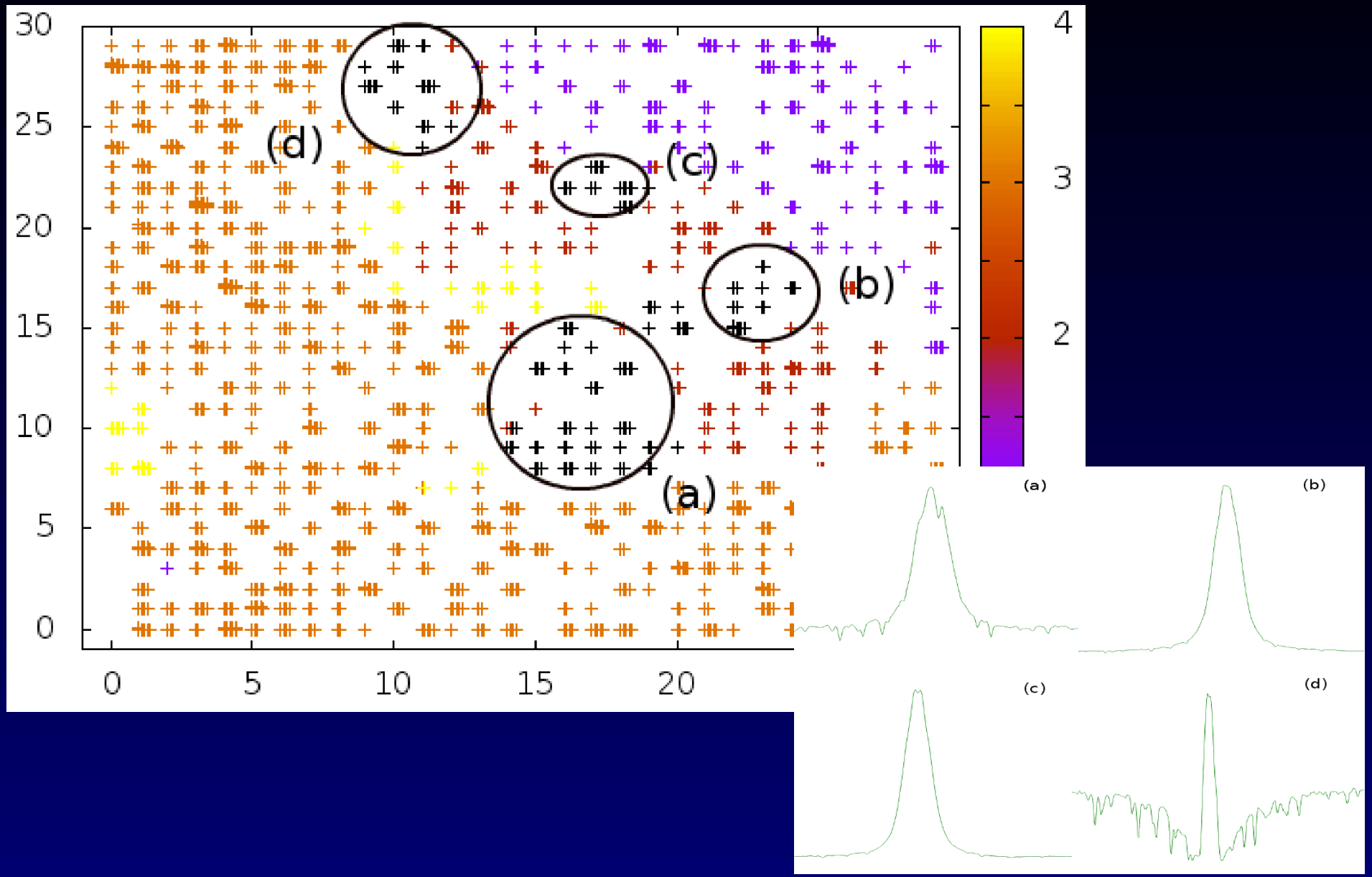
How many vectors activate every neuron

Unified Distance Matrix (U-matrix)

Every neuron= sum of distances to neighbours

The higher = more unique (outlier)

Outliers in clusters



Machine Learning of Spectra
SW view

ML does not produce new data – same spectra in groups

Results the same size as input (+ small overhead)

Tracing visual shape from ML results

- Solf-Organizing maps – finding outliers
- Easy trace shape from neuron - clickable maps
- Visualisation of many spectra in web – dygraph (JS)

Machine Learning of BIG Archive

Idea – 2.2 mil of LAMOST spectra (3.3 mil. in SDSS)

NOT Upload data by user (VO compatible archive)

Driven by SPECTRA LIST (votable obtained by TAP ?)

Workers on same hi-speed network as archive

Calling SSAP + DL always (client on GRID worker ?)

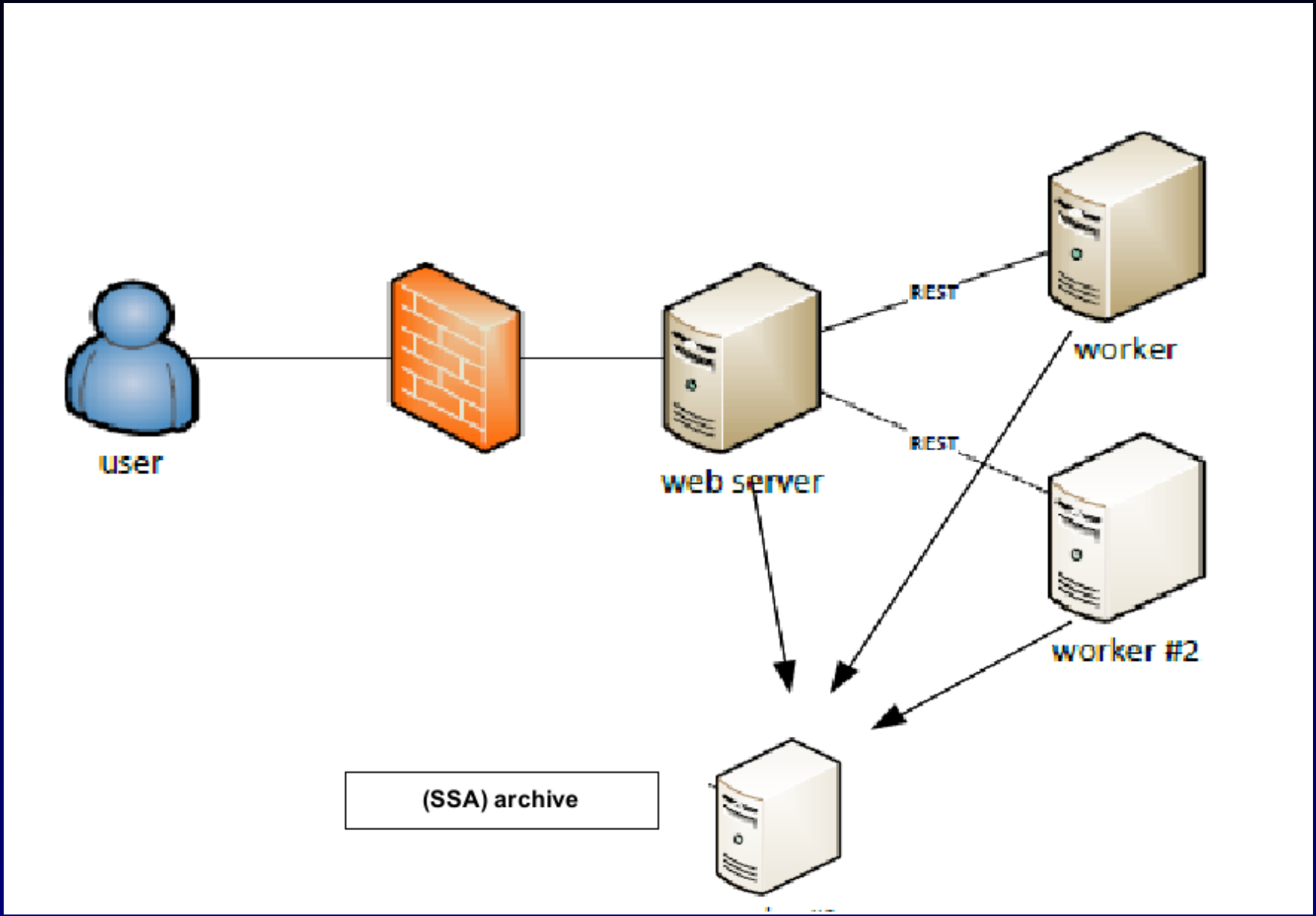
Pre-cache ?

Compute feature vectors – store for whole exeriment ?

PERSISTENT STORAGE - network FS ?

Visualisation - needs input data (spectrum), lists from class

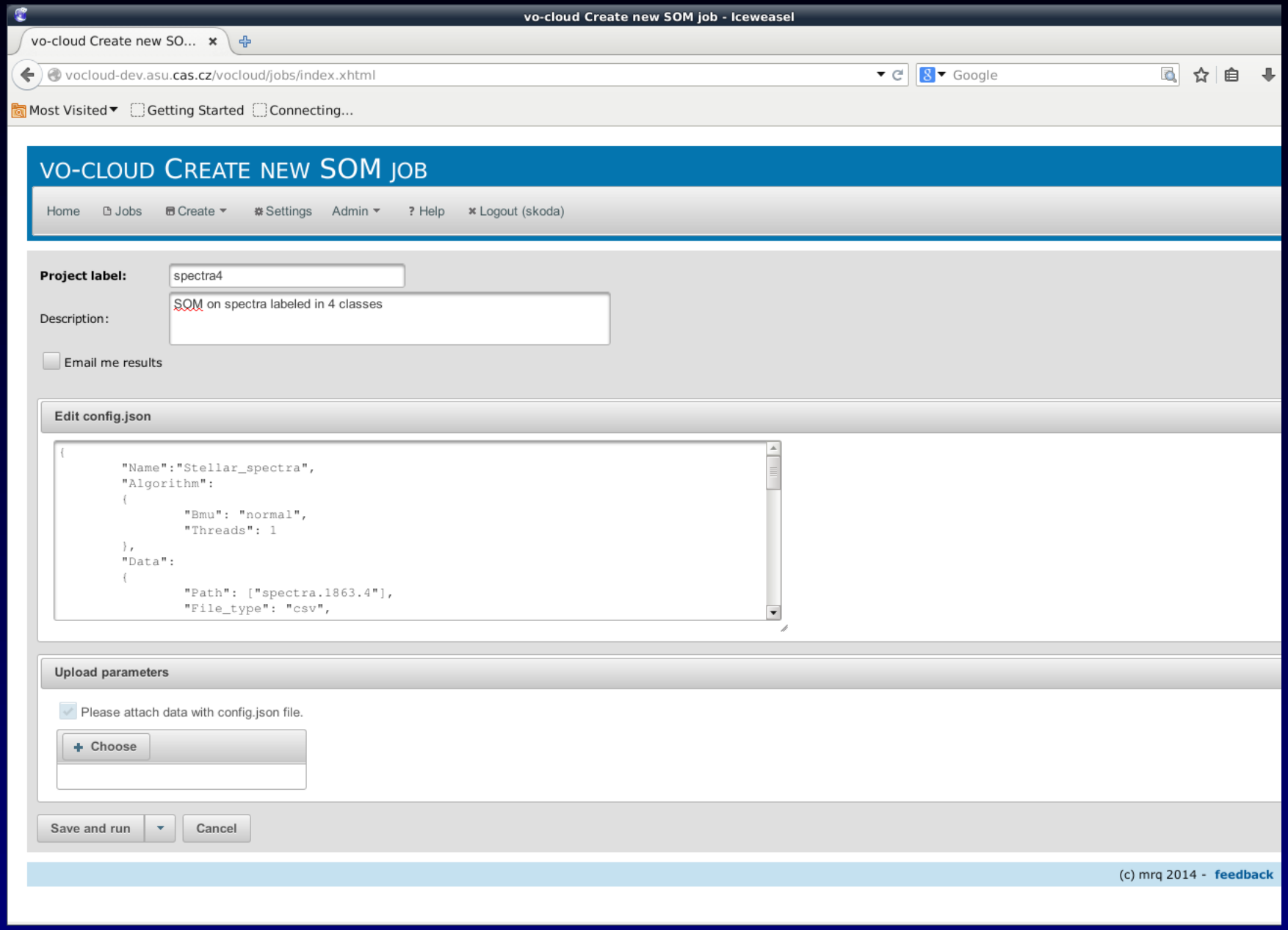
Machine Learning of BIG Archive



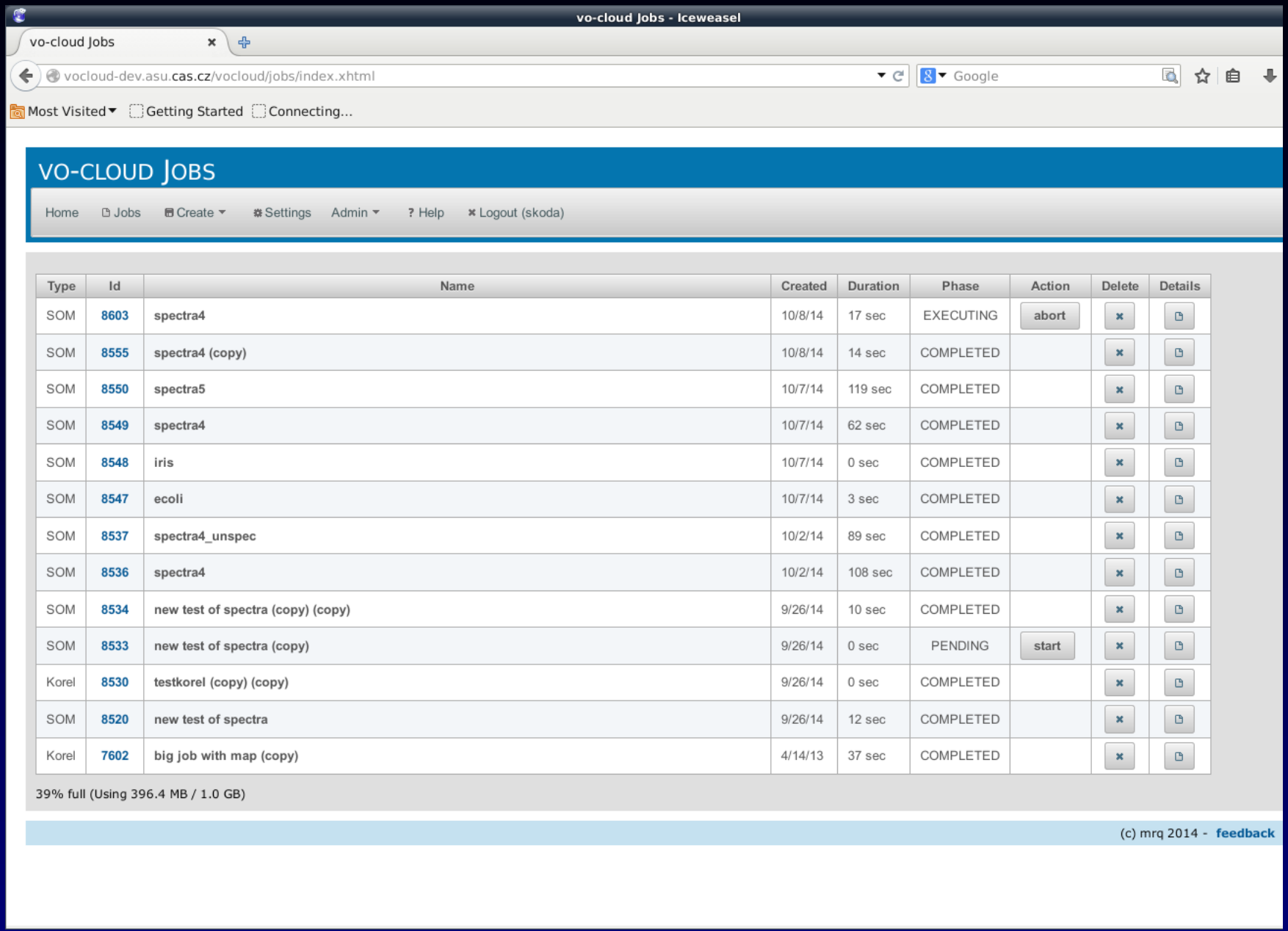
DEMO

http://vocloud-dev.asu.cas.cz

DEMO – create job



DEMO – Job is running



Type	Id	Name	Created	Duration	Phase	Action	Delete	Details
SOM	8603	spectra4	10/8/14	17 sec	EXECUTING	abort	×	
SOM	8555	spectra4 (copy)	10/8/14	14 sec	COMPLETED		×	
SOM	8550	spectra5	10/7/14	119 sec	COMPLETED		×	
SOM	8549	spectra4	10/7/14	62 sec	COMPLETED		×	
SOM	8548	iris	10/7/14	0 sec	COMPLETED		×	
SOM	8547	ecoli	10/7/14	3 sec	COMPLETED		×	
SOM	8537	spectra4_unspec	10/2/14	89 sec	COMPLETED		×	
SOM	8536	spectra4	10/2/14	108 sec	COMPLETED		×	
SOM	8534	new test of spectra (copy) (copy)	9/26/14	10 sec	COMPLETED		×	
SOM	8533	new test of spectra (copy)	9/26/14	0 sec	PENDING	start	×	
Korel	8530	testkorel (copy) (copy)	9/26/14	0 sec	COMPLETED		×	
SOM	8520	new test of spectra	9/26/14	12 sec	COMPLETED		×	
Korel	7602	big job with map (copy)	4/14/13	37 sec	COMPLETED		×	

DEMO – Output part1 (map)

vo-cloud Details of job 8603 - Iceweasel

vo-cloud Details of job 8...

vocloud-dev.asu.cas.cz/vocloud/jobs/index.xhtml

Most Visited | Getting Started | Connecting...

VO-CLOUD DETAILS OF JOB 8603

Home | Jobs | Create | Settings | Admin | ? Help | Logout (skoda)

spectra4

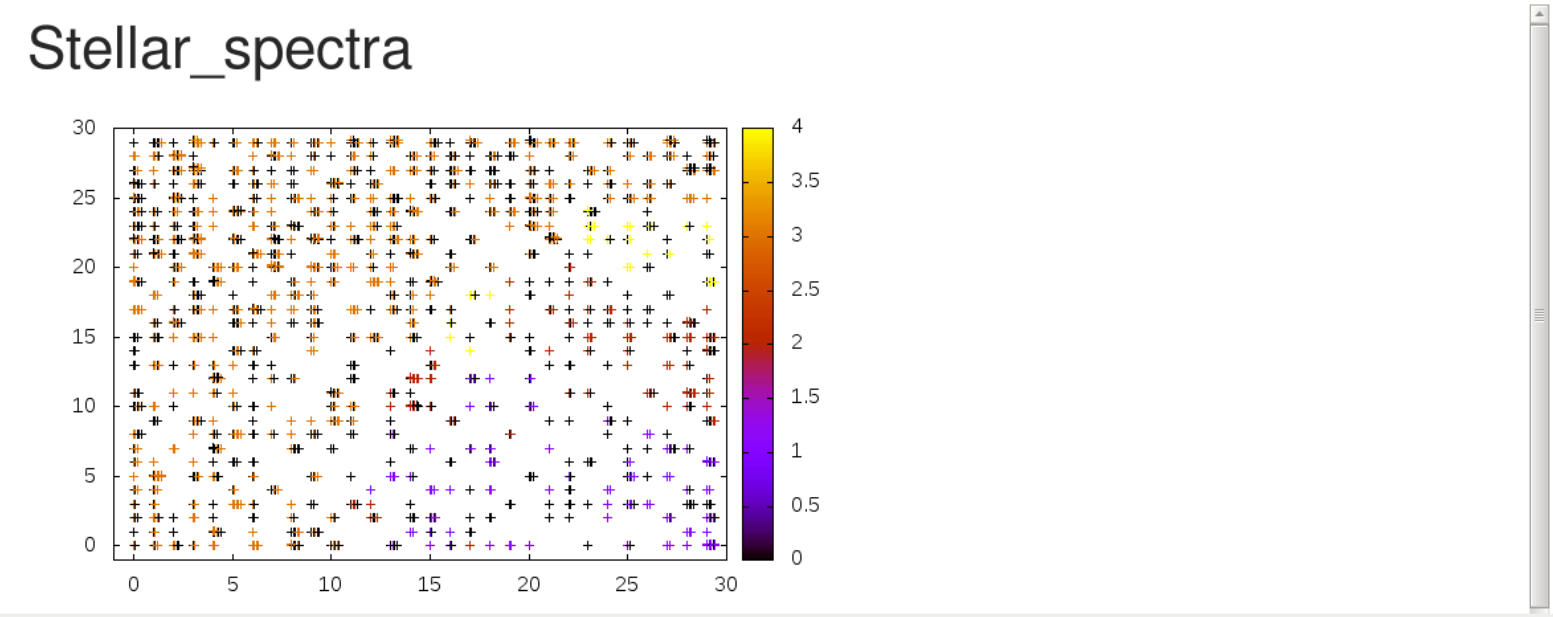
SOM on spectra labeled in 4 classes

Type	Id	Phase	Worker	Created	Started	Finished	Executing time
SOM	8603	COMPLETED	som local	10/8/14 2:31:27 PM	10/8/14 2:31:31 PM	10/8/14 2:32:45 PM	73 sec

run again | delete

Preview

index.html - fullscreen



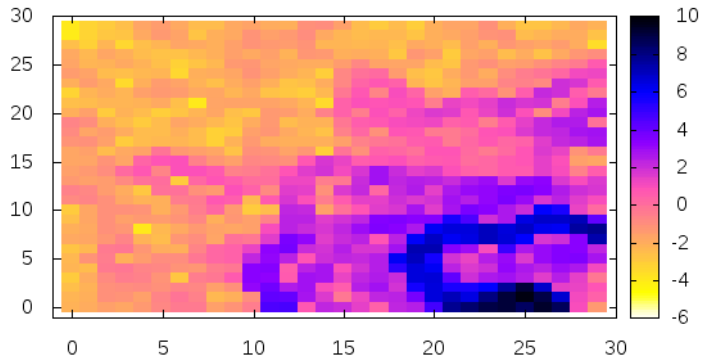
DEMO – Output Part 2 - U-matrix

vo-cloud Details of job 8603 - Iceweasel

vo-cloud Details of job 8...

vocloud-dev.asu.cas.cz/vocloud/jobs/index.xhtml

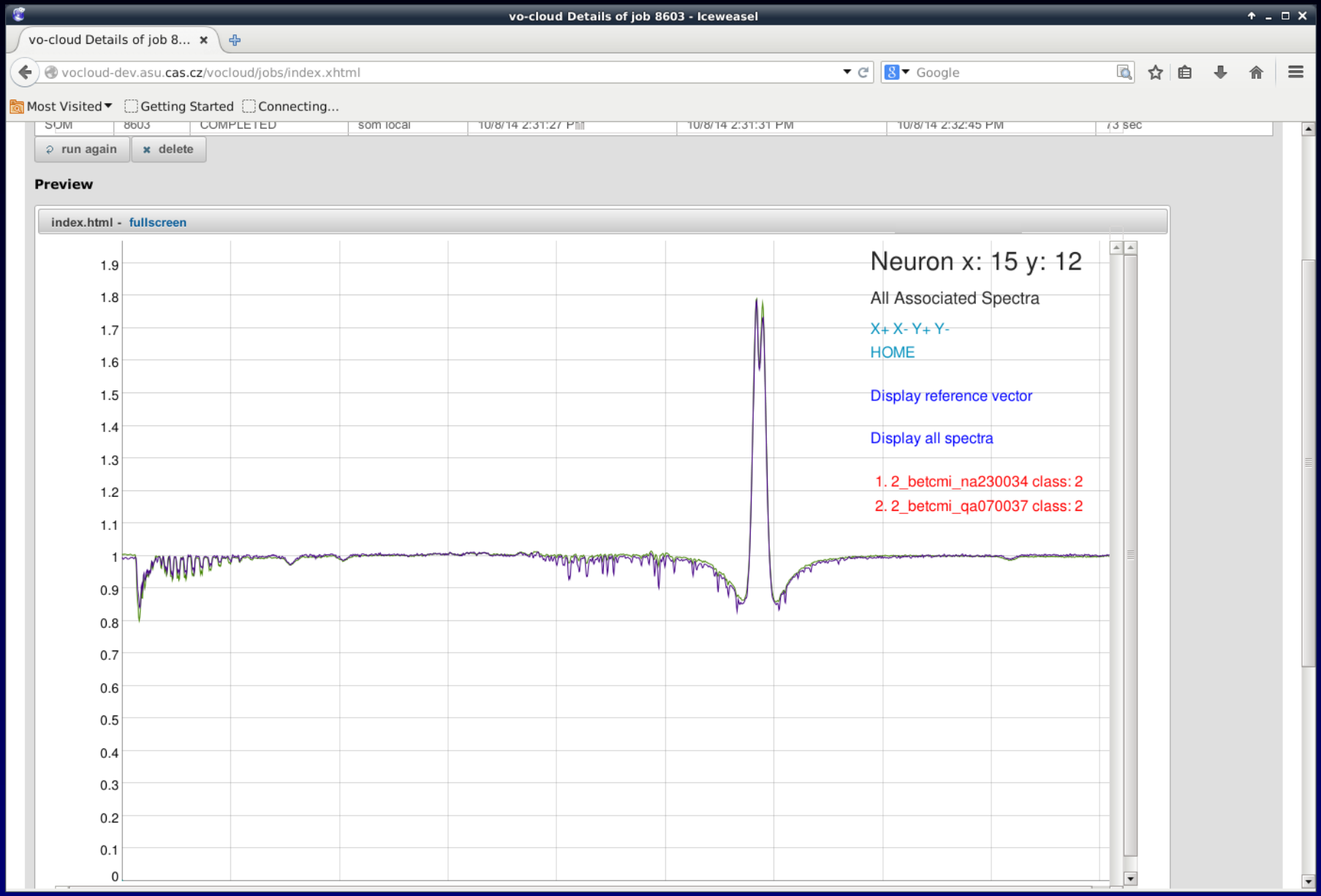
Most Visited | Getting Started | Connecting...



Files

Name	Size	Download	View
config.json	691		
spectra4_classes.zip	9033994		
uws-job.xml	1046		
results.zip	15817894		
errors.txt	158		
run.out	2362		

DEMO


vo-cloud Details of job 8603 - Iceweasel
vo-cloud Details of job 8...
vocloud-dev.asu.cas.cz/vocloud/jobs/index.xhtml
Google
Most Visited
Getting Started
Connecting...
SOM
8603
COMPLETED
som local
10/8/14 2:31:27 PM
10/8/14 2:31:31 PM
10/8/14 2:32:45 PM
73 sec
run again
delete
Preview
index.html - fullscreen
Neuron x: 15 y: 12
All Associated Spectra
X+ X- Y+ Y-
HOME
Display reference vector
Display all spectra
1. 2_betcmi_na230034 class: 2
2. 2_betcmi_qa070037 class: 2
1.9
1.8
1.7
1.6
1.5
1.4
1.3
1.2
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0