AiN

Heidelberg Institute for
Theoretical Studies

HITS

ESCAPE to Victory

Building the infrastructure for the next generation astronomy

ESCAPE

European Science Cluster of Astronomy &
Particle physics ESFRI research Infrastructures

Challenges

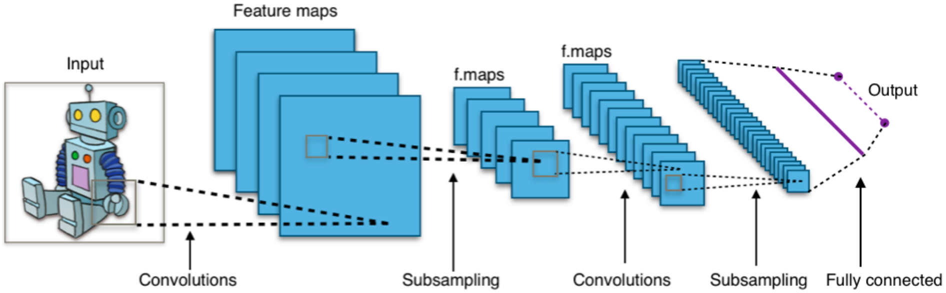
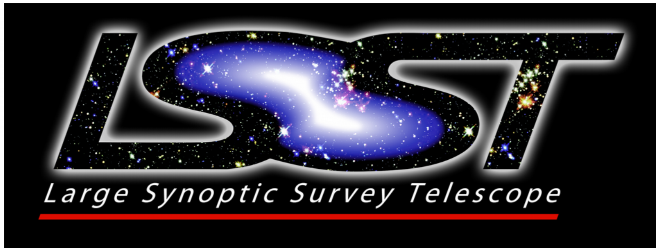
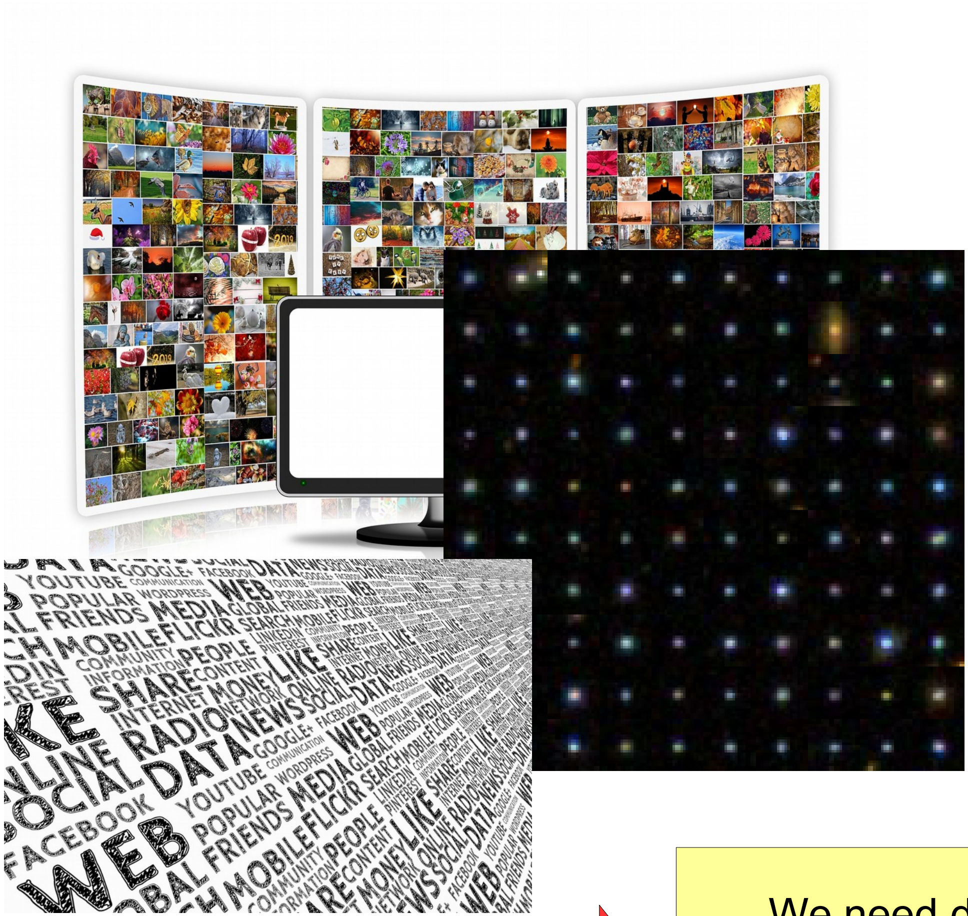


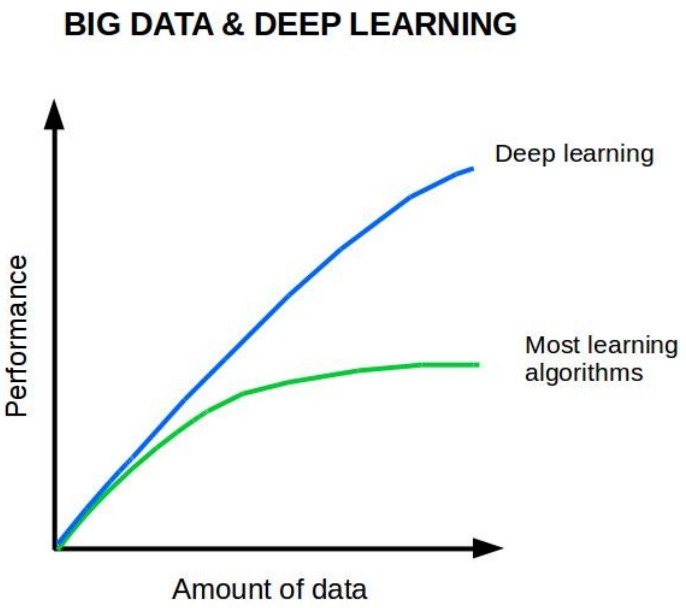
Image from Wikipedia

Big Data by Nick Youngson CC BY-SA 3.0 Alpha Stock Images



We are hungry for data!





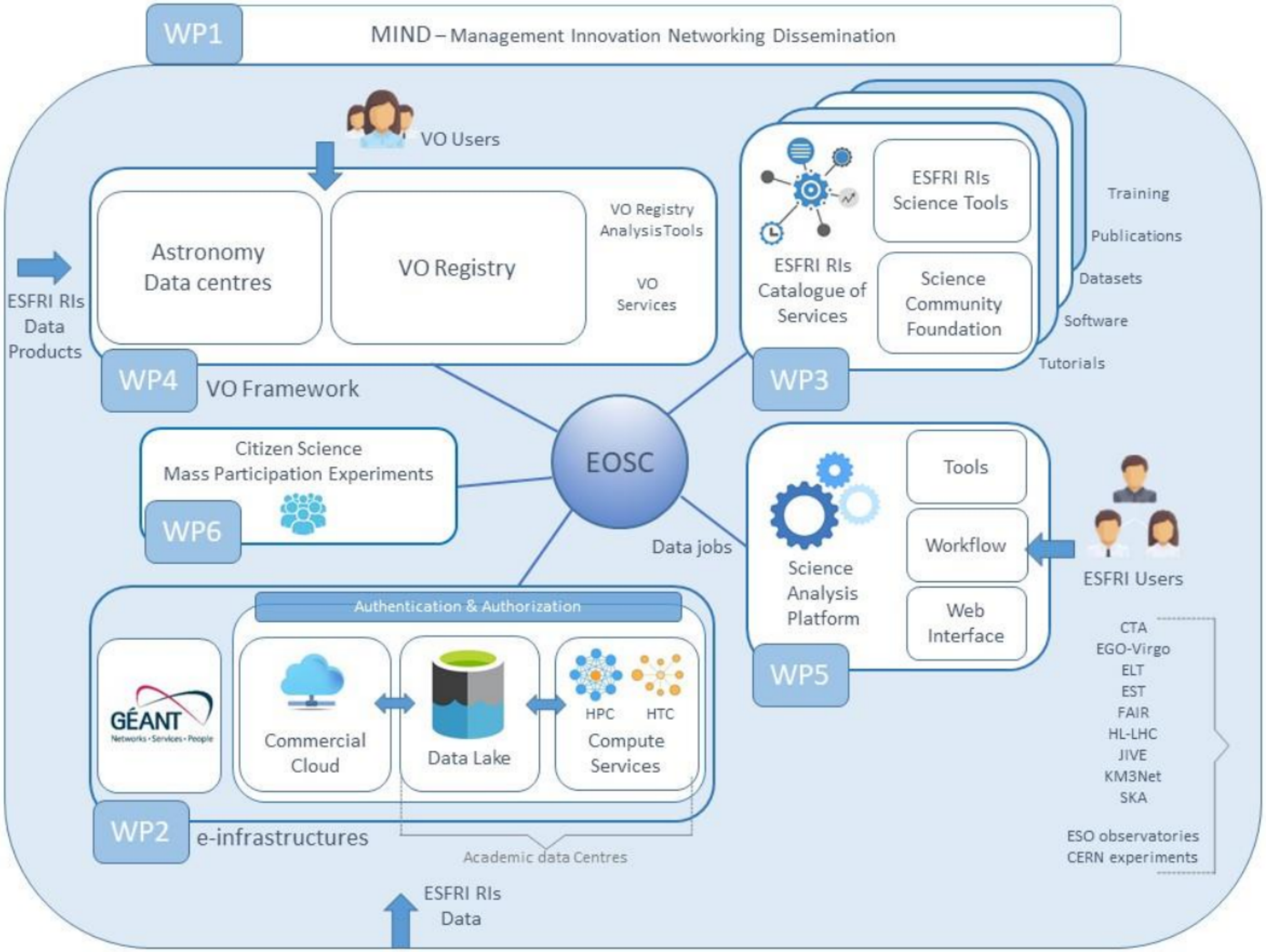
We need data,
we need standards,
we need tools!

ESCAPE project

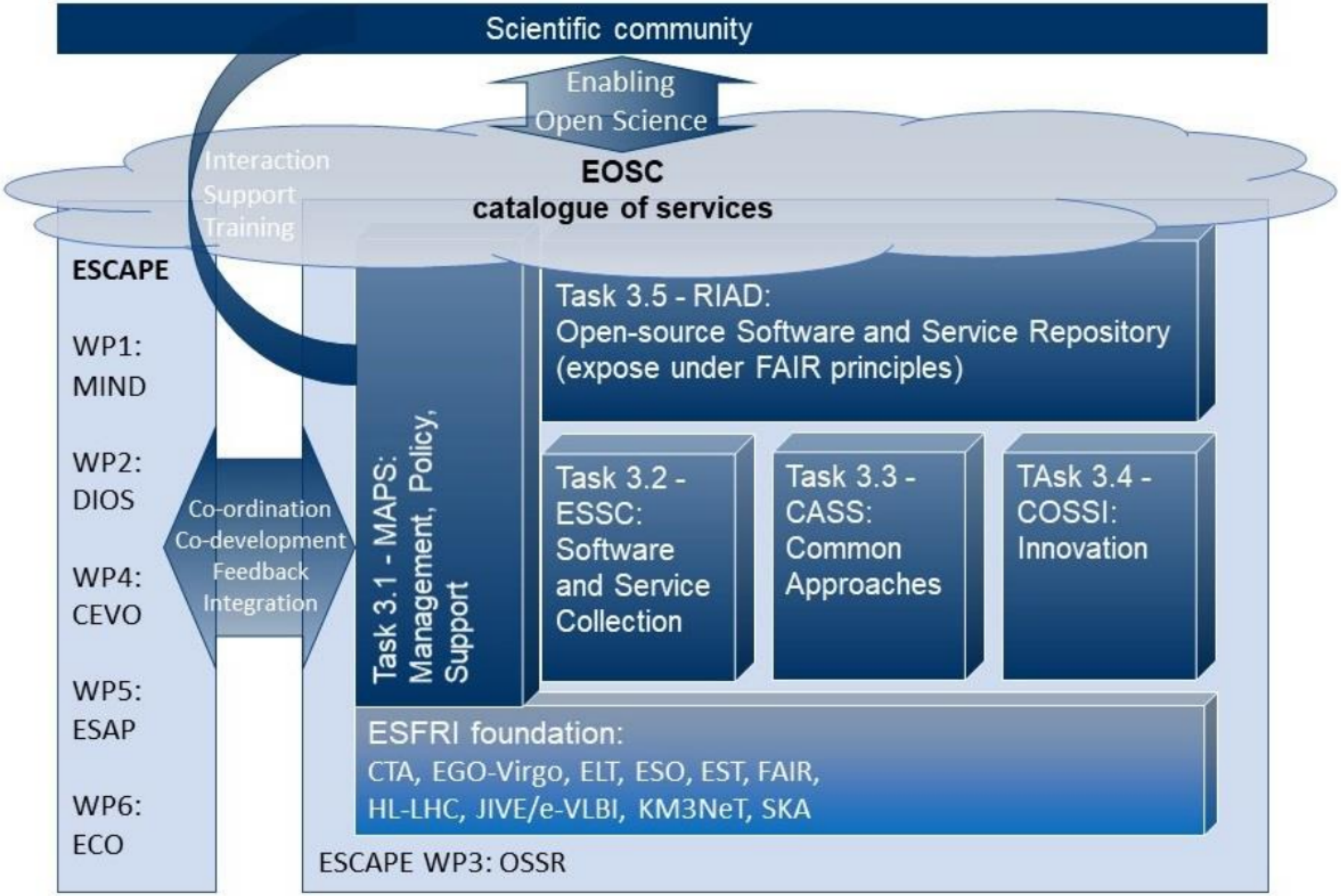
ESCAPE: European Science Cluster of Astronomy & Particle physics ESFRI research infrastructures

- Accessibility to huge amount of data provided by research infrastructures and facilities
- Bring together partners from astronomy and particle physics
- Deliver solutions to ensure integration of data, tools, services and software
- Build standards and ensure interoperability

Project overview



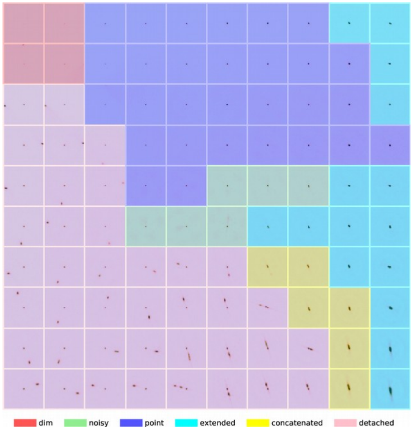
WP3

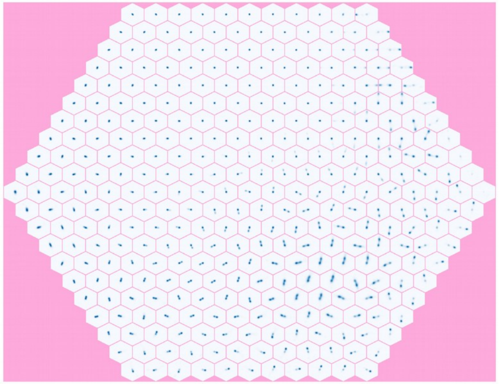

Scientific community
Enabling
Open Science
Interaction
Support
Training
EOSC
catalogue of services
ESCAPE
WP1:
MIND
WP2:
DIOS
WP4:
CEVO
WP5:
ESAP
WP6:
ECO
Co-ordination
Co-development
Feedback
Integration
Task 3.1 - MAPS:
Management, Policy,
Support
Task 3.5 - RIAD:
Open-source Software and Service Repository
(expose under FAIR principles)
Task 3.2 -
ESSC:
Software
and Service
Collection
Task 3.3 -
CASS:
Common
Approaches
TAsk 3.4 -
COSSI:
Innovation
ESFRI foundation:
CTA, EGO-Virgo, ELT, ESO, EST, FAIR,
HL-LHC, JIVE/e-VLBI, KM3NeT, SKA
ESCAPE WP3: OSSR

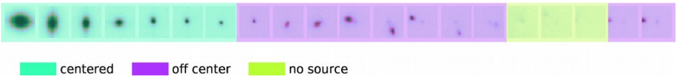
Dimensionality reduction and visualization

The story so far...

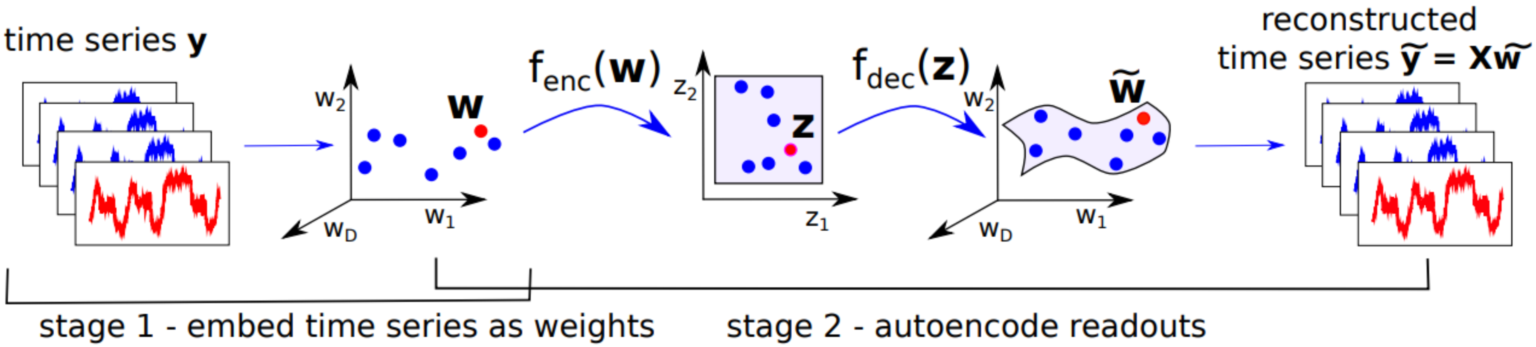
FIRST





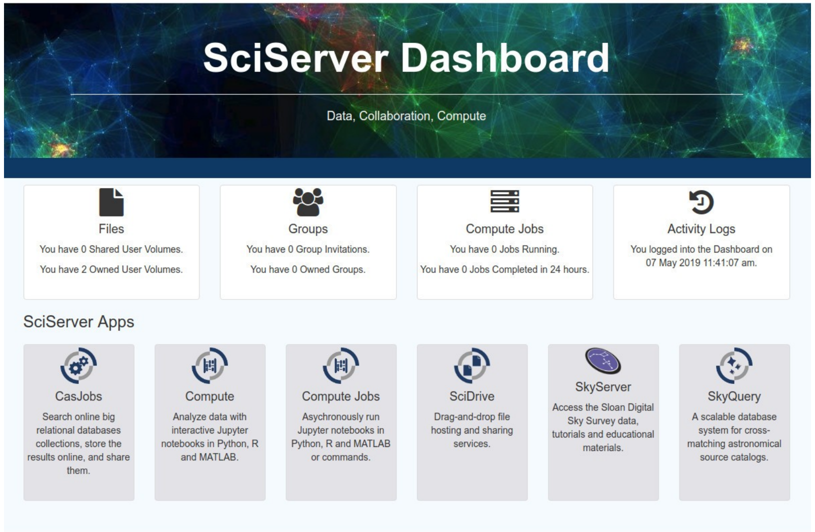


UKIDSS



Challenges: data products

Working with catalogs is a simple task:



SciServer Dashboard

Data, Collaboration, Compute

Files
You have 0 Shared User Volumes.
You have 2 Owned User Volumes.

Groups
You have 0 Group Invitations.
You have 0 Owned Groups.

Compute Jobs
You have 0 Jobs Running.
You have 0 Jobs Completed in 24 hours.

Activity Logs
You logged into the Dashboard on 07 May 2019 11:41:07 am.

SciServer Apps

CasJobs
Search online big relational databases collections, store the results online, and share them.

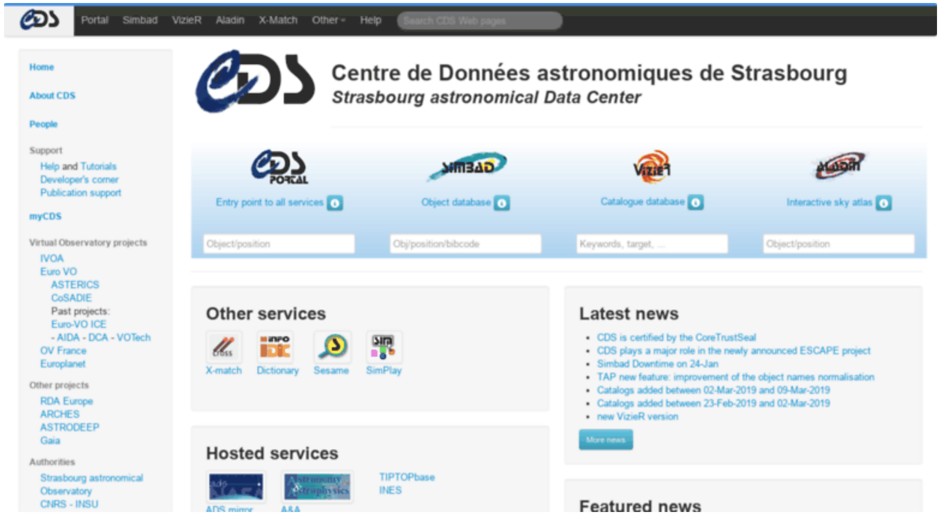
Compute
Analyze data with interactive Jupyter notebooks in Python, R and MATLAB.

Compute Jobs
Asychronously run Jupyter notebooks in Python, R and MATLAB or commands.

SciDrive
Drag-and-drop file hosting and sharing services.

SkyServer
Access the Sloan Digital Sky Survey data, tutorials and educational materials.

SkyQuery
A scalable database system for cross-matching astronomical source catalogs.



Centre de Données astronomiques de Strasbourg

Strasbourg astronomical Data Center

Other services

Hosted services

Latest news

- CDS is certified by the CoreTrustSeal
- CDS plays a major role in the newly announced ESCAPE project
- Simbad Downtime on 24-Jan
- TAP new feature: improvement of the object names normalisation
- Catalogs added between 02-Mar-2019 and 09-Mar-2019
- Catalogs added between 23-Feb-2019 and 02-Mar-2019
- new VizieR version

Featured news

⇨ Problems start with images and spectra!

Some “simple” tasks...

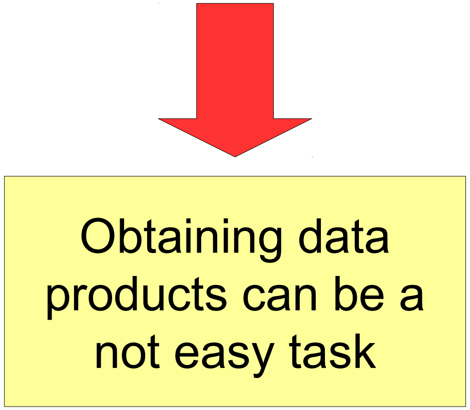
Heidelberg Institute for Theoretical Studies

HITS

1. Given the coordinates, download 28x28 pixel^2 images for all the quasars in SDSS.

2. Download some hundreds of thousands of images from FIRST/UKIDSS.

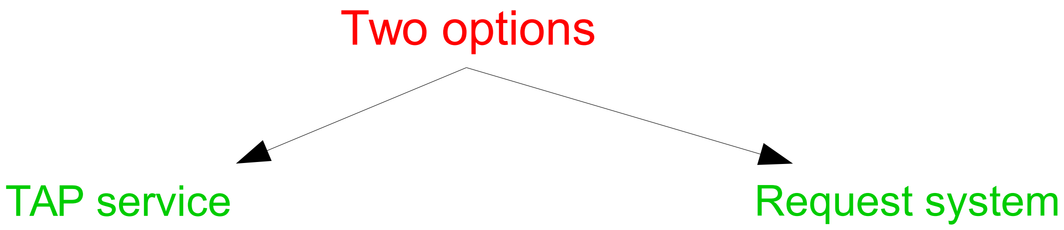
3. Download all the HARPS spectra from ESO archive.



Obtaining data products can be a not easy task

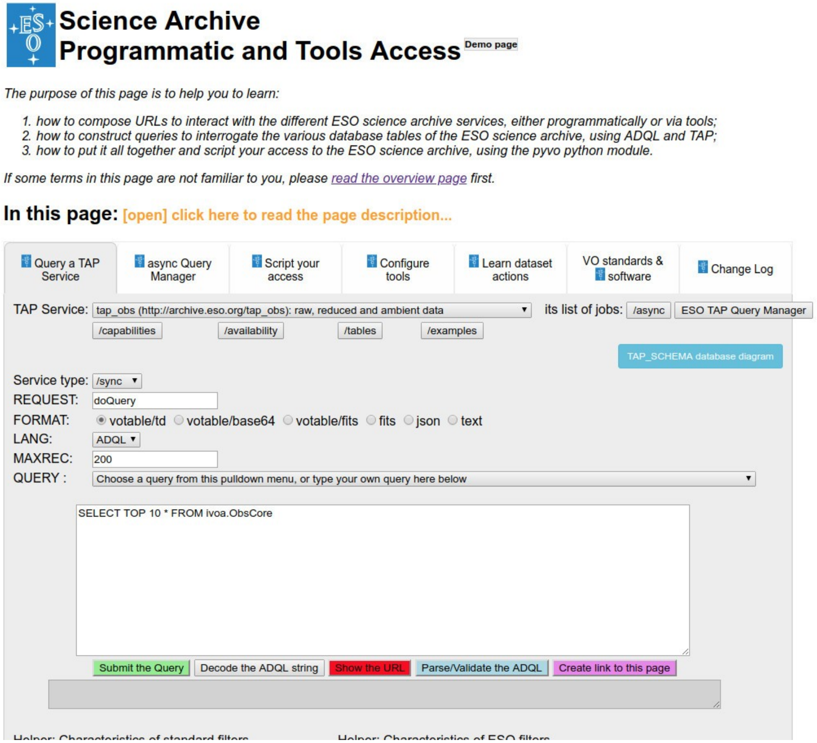
Task 3

Download all the HARPS spectra from ESO archive:


Two options

TAP service

Request system


Science Archive
Programmatic and Tools Access Demo page

The purpose of this page is to help you to learn:

1. *how to compose URLs to interact with the different ESO science archive services, either programmatically or via tools;*
2. *how to construct queries to interrogate the various database tables of the ESO science archive, using ADQL and TAP;*
3. *how to put it all together and script your access to the ESO science archive, using the pyvo python module.*

If some terms in this page are not familiar to you, please read the overview page first.

In this page: [open] click here to read the page description...

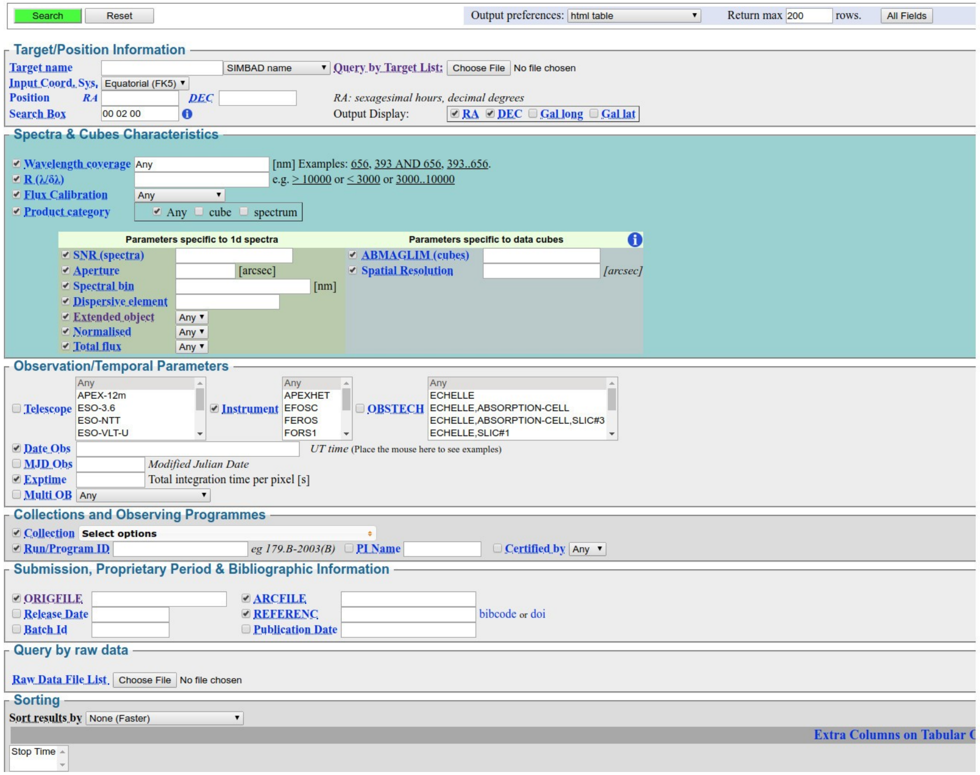
TAP Service: tap_obs (http://archive.eso.org/tap_obs): raw, reduced and ambient data ▾ its list of jobs: /async ESO TAP Query Manager

/capabilities /availability /tables /examples

TAP_SCHEMA database diagram

Service type: /sync ▾
REQUEST: doQuery
FORMAT: ● votable/td ○ votable/base64 ○ votable/fits ○ fits ○ json ○ text
LANG: ADQL ▾
MAXREC: 200
QUERY : Choose a query from this pulldown menu, or type your own query here below ▾

```
SELECT TOP 10 * FROM ivoa.ObsCore
```



Output preferences: html table ▾ Return max 200 rows. All Fields

Target/Position Information

Target name [ ] SIMBAD name ▾ Query by Target List: Choose File No file chosen
Input Coord. Sys. Equatorial (FK5) ▾
Position RA [ ] DEC [ ] *RA: sexagesimal hours, decimal degrees*
Search Box 00 02 00 Output Display: ☑ RA ☑ DEC □ Gal long □ Gal lat

Spectra & Cubes Characteristics

☑ Wavelength coverage Any [nm] Examples: 656, 393 AND 656, 393..656.
☑ R (λ/δλ) e.g. > 10000 or < 3000 or 3000..10000
☑ Flux Calibration Any ▾
☑ Product category ☑ Any □ cube □ spectrum

Parameters specific to 1d spectra		Parameters specific to data cubes	
☑ SNR (spectra)		☑ ABMAGLIM (cubes)	
☑ Aperture	[arcsec]	☑ Spatial Resolution	*[arcsec]*
☑ Spectral bin	[nm]		
☑ Dispersive element			
☑ Extended object	Any ▾		
☑ Normalised	Any ▾		
☑ Total flux	Any ▾		

Observation/Temporal Parameters

□ Telescope: Any, APEX-12m, ESO-3.6, ESO-NTT, ESO-VLT-U
☑ Instrument: Any, APEXHET, EFOSC, FEROS, FORS1
□ OBSTECH: Any, ECHELLE, ECHELLE,ABSORPTION-CELL, ECHELLE,ABSORPTION-CELL,SLIC#3, ECHELLE,SLIC#1

☑ Date Obs [ ] *UT time* (Place the mouse here to see examples)
□ MJD Obs [ ] *Modified Julian Date*
☑ Exptime [ ] Total integration time per pixel [s]
□ Multi OB Any ▾

Collections and Observing Programmes

☑ Collection Select options
☑ Run/Program ID [ ] *eg 179.B-2003(B)* □ PI Name [ ] □ Certified by Any ▾

Submission, Proprietary Period & Bibliographic Information

☑ ORIGFILE ☑ ARCFILE
□ Release Date ☑ REFERENC bibcode or doi
□ Batch Id □ Publication Date

Query by raw data

Raw Data File List. Choose File No file chosen

Sorting

Sort results by None (Faster) ▾

Extra Columns on Tabular C

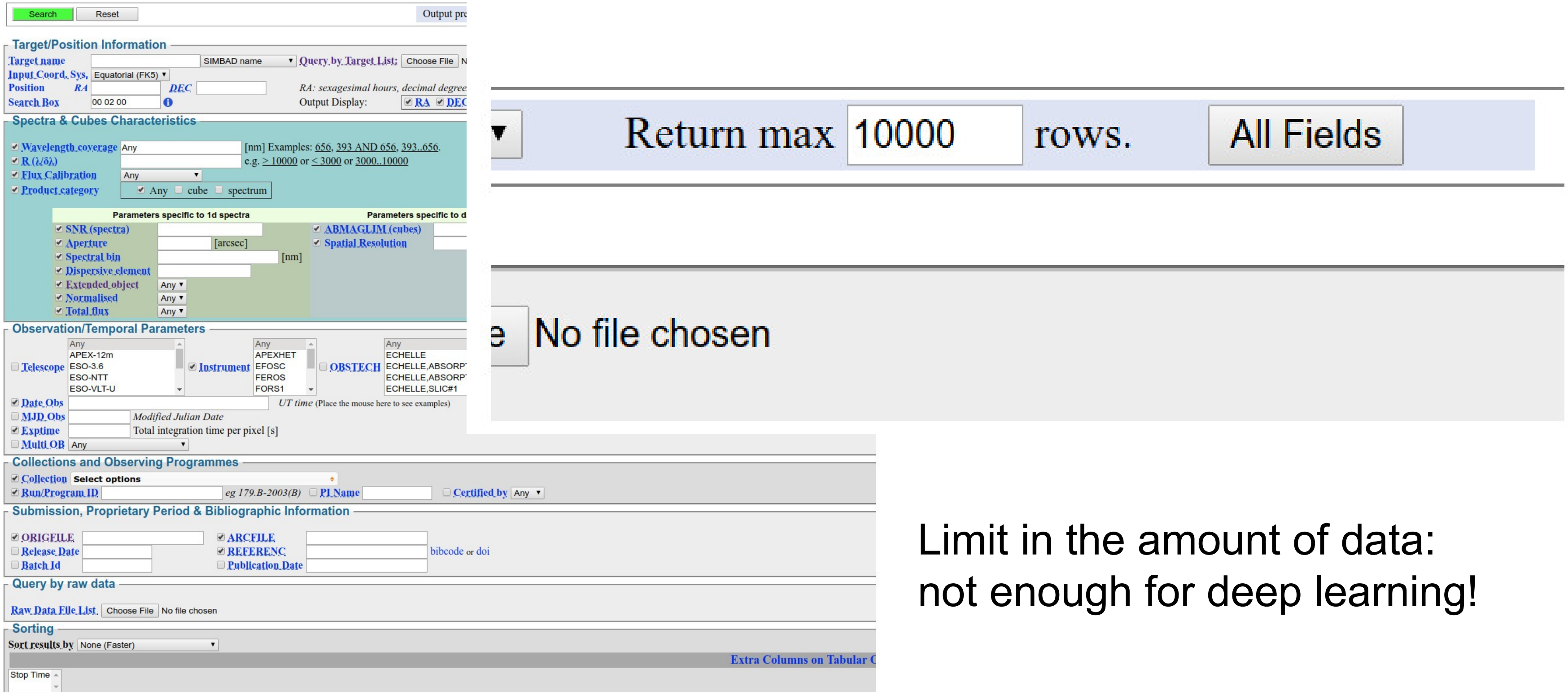
Stop Time

Problems found:

Heidelberg Institute for Theoretical Studies

HITS

Request system



Search Reset Output pre

Target/Position Information

Target name SIMBAD name ▾ Query by Target List: Choose File N

Input Coord. Sys. Equatorial (FK5) ▾

Position RA DEC RA: sexagesimal hours, decimal degree

Search Box 00 02 00 Output Display: ☑ RA ☑ DEC

Spectra & Cubes Characteristics

☑ Wavelength coverage Any [nm] Examples: 656, 393 AND 656, 393..656.

☑ R (λ/δλ) e.g. > 10000 or < 3000 or 3000..10000

☑ Flux Calibration Any ▾

☑ Product category ☑ Any □ cube □ spectrum

Parameters specific to 1d spectra

☑ SNR (spectra)

☑ Aperture [arcsec]

☑ Spectral bin [nm]

☑ Dispersive element

☑ Extended object Any ▾

☑ Normalised Any ▾

☑ Total flux Any ▾

Parameters specific to d

☑ ABMAGLIM (cubes)

☑ Spatial Resolution

Observation/Temporal Parameters

□ Telescope: Any, APEX-12m, ESO-3.6, ESO-NTT, ESO-VLT-U

☑ Instrument: Any, APEXHET, EFOSC, FEROS, FORS1

□ OBSTECH: Any, ECHELLE, ECHELLE,ABSORP, ECHELLE,ABSORP, ECHELLE,SLIC#1

☑ Date Obs UT time (Place the mouse here to see examples)

□ MJD Obs Modified Julian Date

☑ Exptime Total integration time per pixel [s]

□ Multi OB Any ▾

Collections and Observing Programmes

☑ Collection Select options

☑ Run/Program ID eg 179.B-2003(B) □ PI Name □ Certified by Any ▾

Submission, Proprietary Period & Bibliographic Information

☑ ORIGFILE ☑ ARCFILE

□ Release Date ☑ REFERENC bibcode or doi

□ Batch Id □ Publication Date

Query by raw data

Raw Data File List Choose File No file chosen

Sorting

Sort results by None (Faster) ▾

Extra Columns on Tabular O

Stop Time

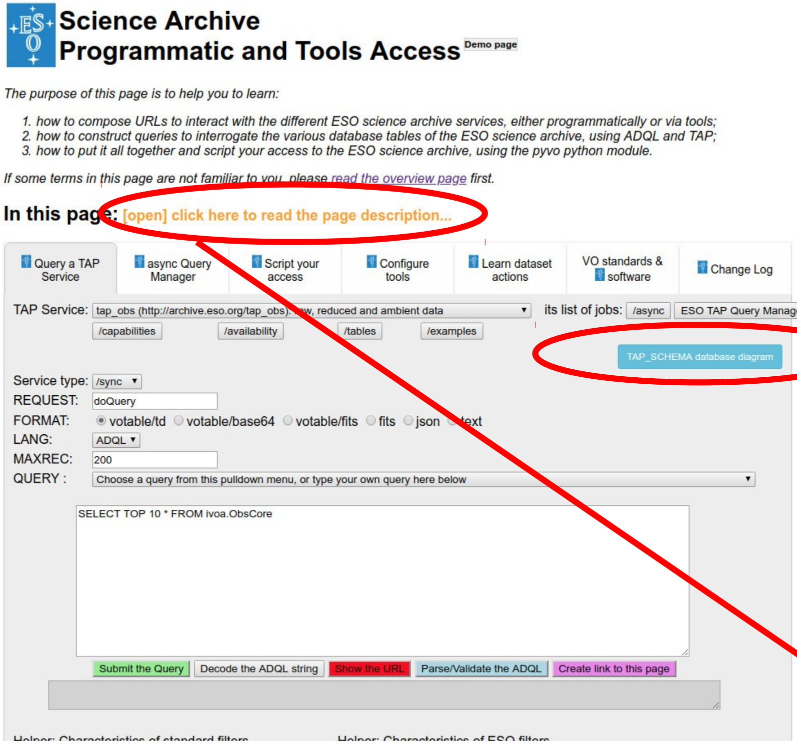
▾ Return max 10000 rows. All Fields

e No file chosen

Limit in the amount of data: not enough for deep learning!

Problems found:

TAP service



Science Archive
Programmatic and Tools Access

Demo page

The purpose of this page is to help you to learn:

1. *how to compose URLs to interact with the different ESO science archive services, either programmatically or via tools;*
2. *how to construct queries to interrogate the various database tables of the ESO science archive, using ADQL and TAP;*
3. *how to put it all together and script your access to the ESO science archive, using the pyvo python module.*

If some terms in this page are not familiar to you, please read the overview page first.

In this page: [open] click here to read the page description...

TAP Service: tap_obs (http://archive.eso.org/tap_obs). raw, reduced and ambient data ▾ its list of jobs: /async ESO TAP Query Manager

Service type: /sync ▾

REQUEST: doQuery

FORMAT: ● votable/td ○ votable/base64 ○ votable/fits ○ fits ○ json ○ text

LANG: ADQL ▾

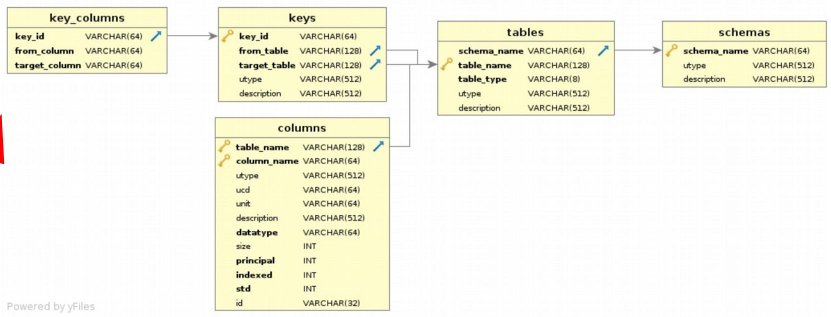
MAXREC: 200

QUERY : Choose a query from this pulldown menu, or type your own query here below ▾

```
SELECT TOP 10 * FROM ivoa.ObsCore
```

Helper: Characteristics of standard filters Helper: Characteristics of ESO filters

- Absence of a clear schema browser



key_columns	
key_id	VARCHAR(64)
from_column	VARCHAR(64)
target_column	VARCHAR(64)

keys	
key_id	VARCHAR(64)
from_table	VARCHAR(128)
target_table	VARCHAR(128)
utype	VARCHAR(512)
description	VARCHAR(512)

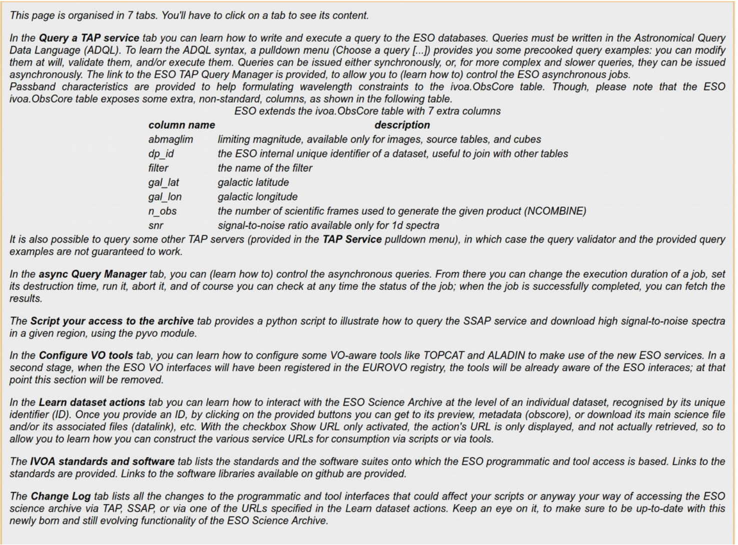
tables	
schema_name	VARCHAR(64)
table_name	VARCHAR(128)
table_type	VARCHAR(8)
utype	VARCHAR(512)
description	VARCHAR(512)

schemas	
schema_name	VARCHAR(64)
utype	VARCHAR(512)
description	VARCHAR(512)

columns	
table_name	VARCHAR(128)
column_name	VARCHAR(64)
utype	VARCHAR(512)
ucd	VARCHAR(64)
unit	VARCHAR(64)
description	VARCHAR(512)
datatype	VARCHAR(64)
size	INT
principal	INT
indexed	INT
std	INT
id	VARCHAR(32)

Powered by yFiles

- Poor and unclear documentation



This page is organised in 7 tabs. You'll have to click on a tab to see its content.

*In the **Query a TAP service** tab you can learn how to write and execute a query to the ESO databases. Queries must be written in the Astronomical Query Data Language (ADQL). To learn the ADQL syntax, a pulldown menu (Choose a query [...]) provides you some precooked query examples: you can modify them at will, validate them, and/or execute them. Queries can be issued either synchronously, or, for more complex and slower queries, they can be issued asynchronously. The link to the ESO TAP Query Manager is provided, to allow you to (learn how to) control the ESO asynchronous jobs.*
Passband characteristics are provided to help formulating wavelength constraints to the ivoa.ObsCore table. Though, please note that the ESO ivoa.ObsCore table exposes some extra, non-standard, columns, as shown in the following table.

ESO extends the ivoa.ObsCore table with 7 extra columns

column name	***description***
abmaglim	*limiting magnitude, available only for images, source tables, and cubes*
dp_id	*the ESO internal unique identifier of a dataset, useful to join with other tables*
filter	*the name of the filter*
gal_lat	*galactic latitude*
gal_lon	*galactic longitude*
n_obs	*the number of scientific frames used to generate the given product (NCOMBINE)*
snr	*signal-to-noise ratio available only for 1d spectra*

*It is also possible to query some other TAP servers (provided in the **TAP Service** pulldown menu), in which case the query validator and the provided query examples are not guaranteed to work.*

*In the **async Query Manager** tab, you can (learn how to) control the asynchronous queries. From there you can change the execution duration of a job, set its destruction time, run it, abort it, and of course you can check at any time the status of the job; when the job is successfully completed, you can fetch the results.*

*The **Script your access to the archive** tab provides a python script to illustrate how to query the SSAP service and download high signal-to-noise spectra in a given region, using the pyvo module.*

*In the **Configure VO tools** tab, you can learn how to configure some VO-aware tools like TOPCAT and ALADIN to make use of the new ESO services. In a second stage, when the ESO VO interfaces will have been registered in the EUROVO registry, the tools will be already aware of the ESO interaces; at that point this section will be removed.*

*In the **Learn dataset actions** tab you can learn how to interact with the ESO Science Archive at the level of an individual dataset, recognised by its unique identifier (ID). Once you provide an ID, by clicking on the provided buttons you can get to its preview, metadata (obscore), or download its main science file and/or its associated files (datalink), etc. With the checkbox Show URL only activated, the action's URL is only displayed, and not actually retrieved, so to allow you to learn how you can construct the various service URLs for consumption via scripts or via tools.*

*The **IVOA standards and software** tab lists the standards and the software suites onto which the ESO programmatic and tool access is based. Links to the standards are provided. Links to the software libraries available on github are provided.*

*The **Change Log** tab lists all the changes to the programmatic and tool interfaces that could affect your scripts or anyway your way of accessing the ESO science archive via TAP, SSAP, or via one of the URLs specified in the Learn dataset actions. Keep an eye on it, to make sure to be up-to-date with this newly born and still evolving functionality of the ESO Science Archive.*

Solutions?

- Python script available on request (thanks Alberto Micol and Martino Romaniello!), but frequent crashes experienced → still investigating the problem, solved by hacking

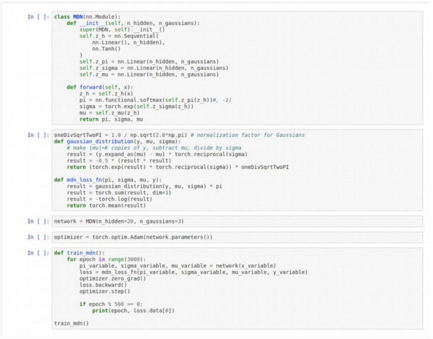
- Download much slower with respect to request system

⇩

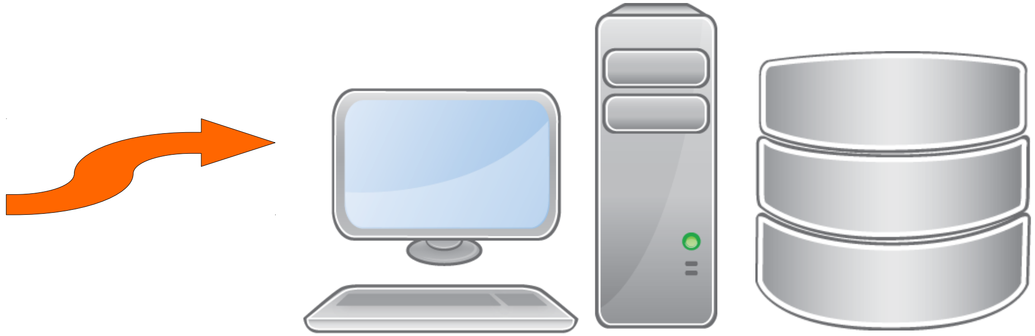
Asking friends and colleagues for data and support can be helpful, but it is not what standardization and the ESCAPE paradigm are about!

Bringing code to the data

Uploading my code and work on server side could solve many issues…



```
In [ ]: class MDN(nn.Module):
            def __init__(self, n_hidden, n_gaussians):
                super(MDN, self).__init__()
                self.z_h = nn.Sequential(
                    nn.Linear(1, n_hidden),
                    nn.Tanh()
                )
                self.z_pi = nn.Linear(n_hidden, n_gaussians)
                self.z_sigma = nn.Linear(n_hidden, n_gaussians)
                self.z_mu = nn.Linear(n_hidden, n_gaussians)

            def forward(self, x):
                z_h = self.z_h(x)
                pi = nn.functional.softmax(self.z_pi(z_h))#, -1)
                sigma = torch.exp(self.z_sigma(z_h))
                mu = self.z_mu(z_h)
                return pi, sigma, mu

In [ ]: oneDivSqrtTwoPI = 1.0 / np.sqrt(2.0*np.pi) # normalization factor for Gaussians
        def gaussian_distribution(y, mu, sigma):
            # make |mu|=K copies of y, subtract mu, divide by sigma
            result = (y.expand_as(mu) - mu) * torch.reciprocal(sigma)
            result = -0.5 * (result * result)
            return (torch.exp(result) * torch.reciprocal(sigma)) * oneDivSqrtTwoPI

        def mdn_loss_fn(pi, sigma, mu, y):
            result = gaussian_distribution(y, mu, sigma) * pi
            result = torch.sum(result, dim=1)
            result = -torch.log(result)
            return torch.mean(result)

In [ ]: network = MDN(n_hidden=20, n_gaussians=3)

In [ ]: optimizer = torch.optim.Adam(network.parameters())

In [ ]: def train_mdn():
            for epoch in range(3000):
                pi_variable, sigma_variable, mu_variable = network(x_variable)
                loss = mdn_loss_fn(pi_variable, sigma_variable, mu_variable, y_variable)
                optimizer.zero_grad()
                loss.backward()
                optimizer.step()

                if epoch % 500 == 0:
                    print(epoch, loss.data[0])

        train_mdn()
```



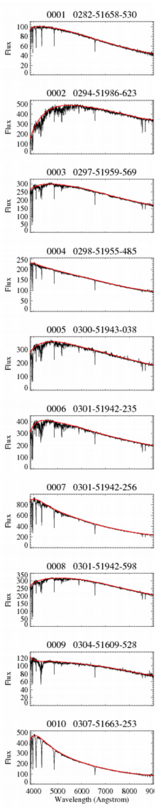
Who is going to provide and pay for resources?

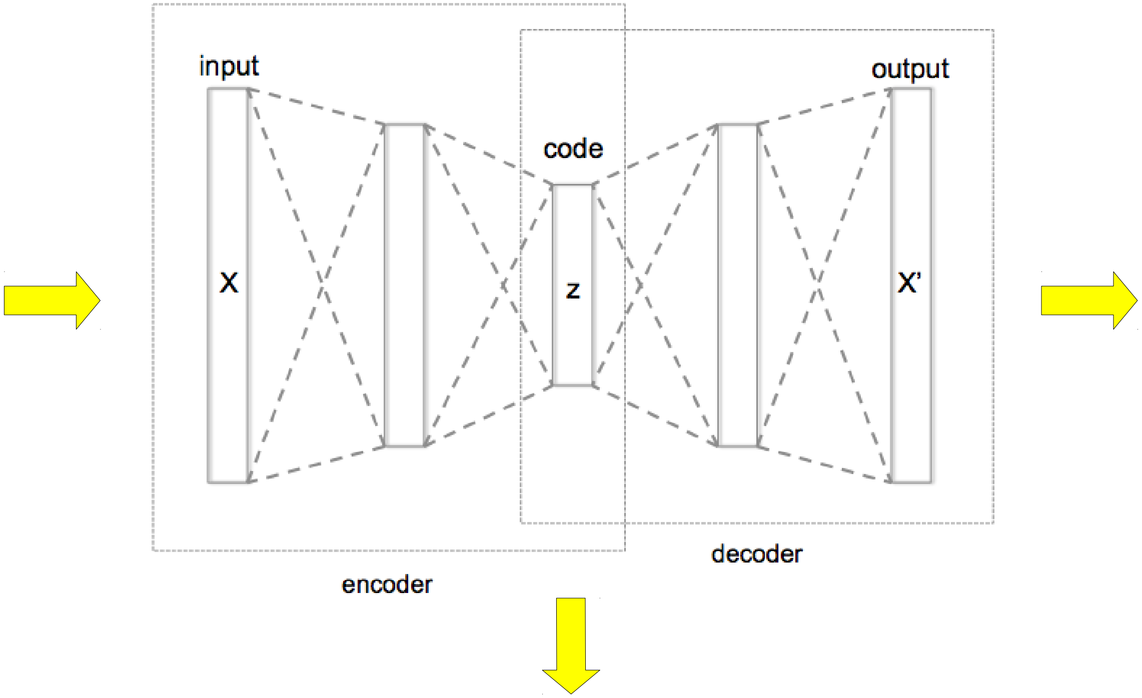
First developments

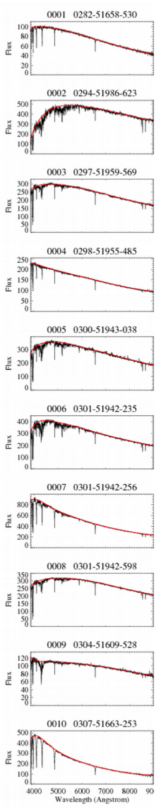
Development of a prototype for:

Dimensionality reduction and analysis of spectra

autoencoder







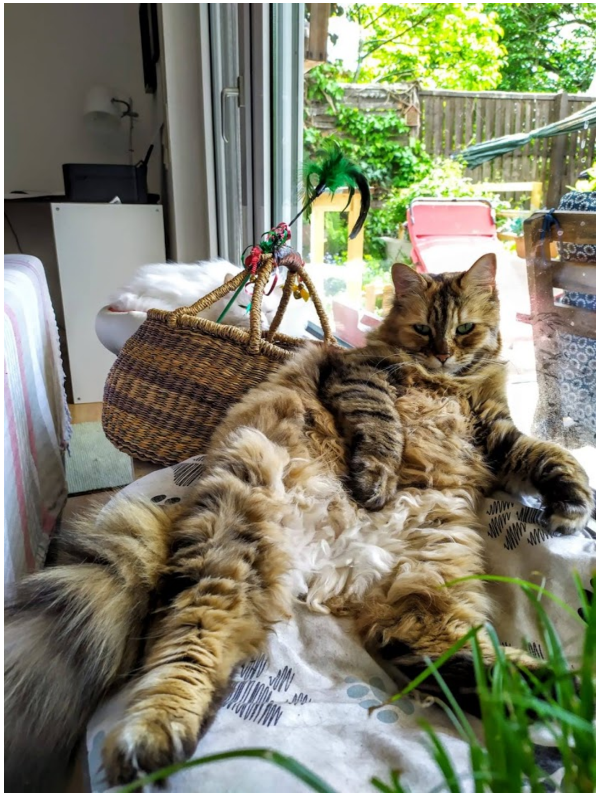
representation

First results

SEE LIVE DEMO

Conclusions

- ESCAPE project is going to be a step to build a new infrastructure for data-intense astronomy
- A lot of work to do:
  - ✗ data products access
  - ✗ building standards
  - ✗ bringing code to the data
- Development of a first prototype → big potential and future integration in web services
- Final question: are we ready for machine learning and big data?


I'm not labeled, but this machine learning stuff is almost as cool as me!

THANKS!
QUESTIONS?

Acknowledgments: ESCAPE project has been funded by the European Union's Horizon 2020 - Grant num. 824064