ESAP as an Analysis Environment

ESAP Training Workshop

Stelios Voutsinas - stv@roe.ac.uk

ESCAPE - The European Science Cluster of Astronomy & Particle Physics ESFRI Research Infrastructures has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement n° 824064.

ESAP - Interactive Data Analysis (IDA)

Goal:

Develop a way for user to find data, software & compute facilities within the ESCAPE ecosystem, and run analysis on the data of interest using this software, in an interactive way.

A few questions that lead to design decisions:

- How to allow users to discover data?
- How to discover & use workflows or software that can analyse this data?
- How to discover & use compute facilities on which to run the above software with the data of interest?
- Software & workflows need to be annotated with all the metadata needed to allow discoverability, but also automation of provisioning at a given facility.
- How to ensure that a workflow runs in a reproducible way?
- How can we automate as much of this as possible?

Funded by the European Union's Horizon 2020 - Grant N° 824064

ESAP - Interactive Data Analysis (IDA)

ESAP's Interactive Data Analysis feature allows users to:

- Find notebooks & software from the ESCAPE community,
- Find Compute Resources on which Interactive Analysis can be performed
- Provision a combination of software/notebook & optionally some data, on the chosen Compute Facility.
- Automate as much of this as possible.
  i.e. Prepare environment for user (Automatically install required libraries & tools)

Funded by the European Union's Horizon 2020 - Grant N° 824064

ESAP - Interactive Data Analysis (IDA)

Initial interest in **workflows** (**Notebook**-based) lead to us focusing on using this as a first target.

ESAP serves as an aggregator of Jupyter-based Services.
It does **not** ship with a JupyterHub implementation.

How to ensure that a Notebook workflow can run in a reproducible way & how can we automate the process of making this available to a user?

> ***BinderHub*** seemed to answer a lot of these questions.
> The list of Notebooks available in the current implementation of ESAP are all Binder-compatible.

Funded by the European Union's Horizon 2020 - Grant N° 824064

BinderHub & Notebooks

As such, first version focuses on:
Notebooks & Binder

The Binder project defines a nice way of packaging a Notebook workflow with all it's required libraries in a way that BinderHub services can prepare the environment for the user.

A Binder repository could contain the following:

- **requirements.txt / environment.yaml** - Required Libraries for the Notebook
- **postBuild** - A script to run after environment is created
- **A content file** - index.ipynb (Example)

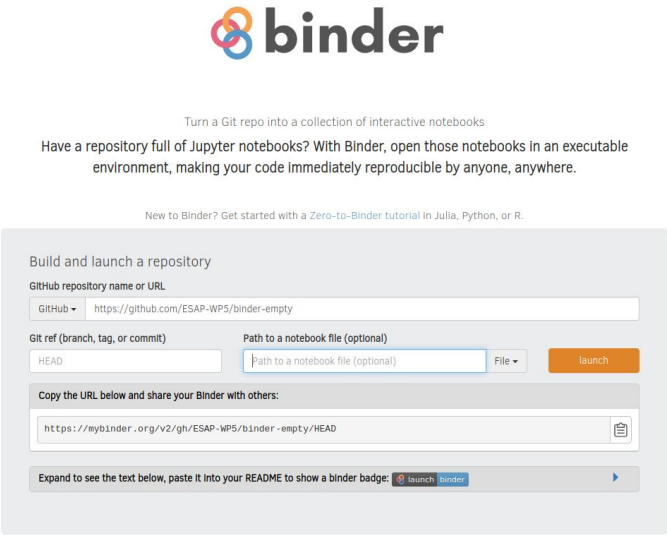
Several other options, for example: Data directory or any other files that could be used by the notebook.

Funded by the European Union's Horizon 2020 - Grant N° 824064

Binder & Notebooks

The way Binder works is:

- Select a repository to use (Optionally a Git ref, Path) and ***Launch***
- Binder creates Docker container (unless it already exists in repo), in which required libraries are installed
- User is then redirected to a Jupyter Lab page with the notebook which should be run in a reproducible way.
- This redirect URL can be generated programmatically, and this is what we do in ESAP



Funded by the European Union's Horizon 2020 - Grant N° 824064

Finding Software & Notebooks

First step in an IDA access scenario is to find software & notebooks

First release limited to notebooks, but designed to support other tools for Data Analysis CLI or Desktop-based

Discovery page to allow users to search by:

- Keyword - Match keyword string against all fields of each software record (description, name, author etc..)
- Author - Search for software from specific authors
- User Interface Type - CLI Tools, Desktop, Notebooks
- Runtime Platform - Python, Java, etc..

Funded by the European Union's Horizon 2020 - Grant N° 824064

ESAP - Finding Software & Notebooks

ESCAPE ESAP | Archives | Multi Query | Interactive Analysis | Batch Analysis | Asynchronous Jobs | IVOA-SAMP

Interactive Analysis

Notebooks

Search for Workflows

Advanced Search

CSIC-IAA HCG-16 workflow

Description: Analysis of Hickson Compact Group 16
Link: https://github.com/AMIGA-IAA/hcg-16
Author:
Runtime Platform: Python
Keywords: jupyter-notebook

CDS MOCPy

Description: Experiment with Multi-Order Coverage maps
Link: https://github.com/cds-astro/mocpy
Author:
Runtime Platform: Python
Keywords: jupyter-notebook

ESCAPE ESAP | Archives | Multi Query | Interactive Analysis | Batch Analysis | Asynchronous Jobs | IVOA-SAMP

Interactive Analysis

Notebooks

Search for Workflows

Advanced Search

Search By:

User Interface type: All
Author: Author name
Runtime Platform: All

Search

Funded by the European Union's Horizon 2020 - Grant N° 824064

ESAP - What software is available?

- Software & notebooks from the OSSR (ESCAPE Software Repository)
- Notebooks provided by ESAP (included in ESAP DB)
  - For example, blank notebook, or Apertif notebook used to demonstrate access to ESAP shopping client & visualization techniques
- Desktop tools (i.e. Topcat) provided by ESAP (Hidden in initial release)

Funded by the European Union's Horizon 2020 - Grant N° 824064

OSSR & ESAP Integration

OSSR:

- Open-source scientific Software and Service Repository (OSSR)
- Implementation as a Zenodo repository

How do we interact with it?
The OSSR/Zenodo has an API which allows programmatic access to its records
We use a python client provided by WP3 https://gitlab.in2p3.fr/escape2020/wp3/eossr

We harvest the following fields for each entry:

- **ID**
- **Description**
- **URL**
- **RuntimePlatform**

Zooniverse: Advanced Project Building

Description: Demonstrates techniques for advanced Zooniverse project management using Python.
Link: https://git.astron.nl/astron-sdc/escape-wp5/workflows/zooniverse-advanced-project-building
Author: Hugh Dickinson
Runtime Platform: Python
Keywords: jupyter-notebook

This metadata appears for users for better discovery
Also used by us for things like association with compute / filtering
List we get from OSSR contains many more fields, possible to extend

Funded by the European Union's Horizon 2020 - Grant N° 824064

ESAP - Finding Compute Facilities

Once a given software & notebook record has been selected, a list of compute facilities is shown

The list shown, depends on the type of Software that has been selected

- Notebook -> BinderHub facilities
- Desktop/CLI tools -> Rosetta facilities (Disabled until implementation is completed)

Users can filter again using keyword matching

Funded by the European Union's Horizon 2020 - Grant N° 824064

ESAP - Finding Compute Facilities

ESCAPE ESAP | Archives | Multi Query | Interactive Analysis | Batch Analysis | Asynchronous Jobs | IVOA-SAMP

Interactive Analysis

Compute Facilities

Search for Facilities

« Deploy

JIVE BinderHub

Description:
JIVE BinderHub
Link: http://jupyter.jive.nl/binderhub/

MyBinder

Description:
MyBinder
Link: https://mybinder.org/

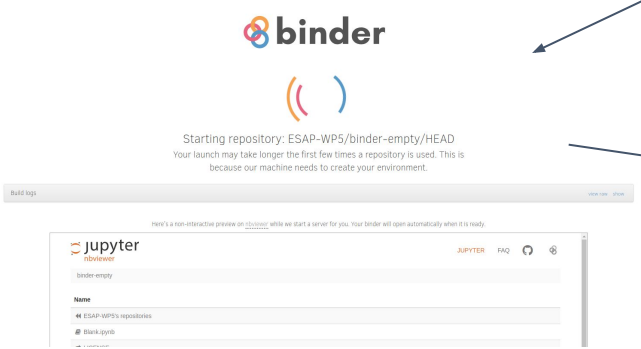
Funded by the European Union's Horizon 2020 - Grant N° 824064

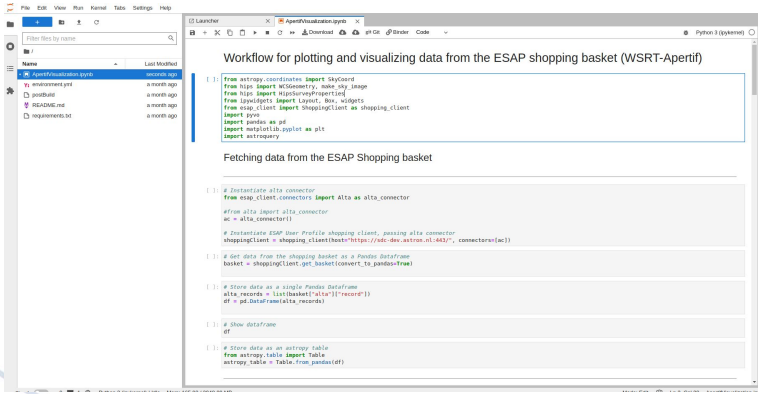
ESAP - Deploy

Params: [Compute Facility, Notebook Repo]

ESAP: Deployer

(Generate Redirect URL..)





Funded by the European Union's Horizon 2020 - Grant N° 824064

ESAP - Accessing ESAP Data in IDA environments

One of the goals was to make Data that has been selected through ESAP available in an IDA environment.

Currently possible via the use of the ESAP shopping client (Python)

- Data (found in Query pages) gets added to the shopping basket
- Notebook & Facility is selected and user is redirected to Jupyter session.
- User can then import the ESAP shopping client and fetch the data from ESAP (via API requests)
- Future iterations may make this process automated

Funded by the European Union's Horizon 2020 - Grant N° 824064

ESAP - Accessing ESAP Data in IDA environments

ESCAPE ESAP ESFRI Science Analysis Platform

Archives | Multi Query | Interactive Analysis | Batch Analysis | Asynchronous Jobs | IVOA-SAMP

Data Shopping Basket

Empty Basket | API (expert user)

Basket	Source	Item
☑	vo_reg	{"archive":"vo_reg","record":{"image":"continuum","3":"apertif-dr1","190807041":"190807041_AP_B001","ivo://astron.nl/~?APERTIF_DR1/190807041_AP_B001/image_mf_02.fits":"https://vo.astron.nl/getproduct/APERTIF_DR1/190807041_AP_B001/image_mf_02.fits","37006":"m1403+5324","208.3603787 ICRS 211.0482971187 50.5831193154 211.2640041493 53.996220402 205.4549338507 53.996220402 205.6706408813 50.5831193154","3.999998420476911":"58702.6901608794","58702.6901608794":"nan","nan":"0.209645077586174","0.23421286046505":"nan","em.radio.750-1500MHz":"/I/","WSRT":"Apertif","3073":"3073","-1":"-1","nan":"nan","https://vo.astron.nl/getproduct/APERTIF_DR1/190807041_AP_B001/image_mf_02.fits?preview=True":"apertif_dr1.continuum_images

ESAP-GUI version Tue Nov 8 15:01:36 2022 +0000

Description: Experiment with Multi-Order Coverage maps
Link: https://github.com/cds-astro/mocpy
Author:
Runtime Platform: Python
Keywords: jupyter-notebook

ASTRON VO Apertif

Description: ASTRON VO Apertif
Link: https://git.astron.nl/astron-sdc/escape-wp5/workflows/apertif-vo-example
Author:
Runtime Platform: Python
Keywords: jupyter-notebook

Apertif Visualization ☑

Description: Shopping Basket and Apertif Visualization Example
Link: https://git.astron.nl/astron-sdc/escape-wp5/workflows/apertif-vis-example
Author:
Runtime Platform: Python
Keywords: jupyter-notebook

ESCAPE ESAP ESFRI Science Analysis Platform

Archives | Multi Query | Interactive Analysis | Batch Analysis | Asynchronous Jobs | IVOA-SAMP

Interactive Analysis

Compute Facilities

Search for Facilities

« | Deploy

JIVE BinderHub

Description:
JIVE BinderHub
Link: http://jupyterjive.nl/binderhub/

MyBinder ☑

Description:
MyBinder
Link: https://mybinder.org/

File Edit View Run Kernel Tabs Settings Help

Filter files by name

Name	Last Modified
ApertifVisualization.ipynb	seconds ago
environment.yml	a month ago
postBuild	a month ago
README.md	a month ago
requirements.txt	a month ago

Launcher | ApertifVisualization.ipynb

Download | Git | Binder | Code | Python 3 (ipykernel)

Workflow for plotting and visualizing data from the ESAP shopping basket (WSRT-Apertif)

```
[ ]: from astropy.coordinates import SkyCoord
     from hips import WCSGeometry, make_sky_image
     from hips import HipsSurveyProperties
     from ipywidgets import Layout, Box, widgets
     from esap_client import ShoppingClient as shopping_client
     import pyvo
     import pandas as pd
     import matplotlib.pyplot as plt
     import astroquery
```

Fetching data from the ESAP Shopping basket

```
[ ]: # Instantiate alta connector
     from esap_client.connectors import Alta as alta_connector

     #from alta import alta_connector
     ac = alta_connector()

     # Instantiate ESAP User Profile shopping client, passing alta connector
     shoppingClient = shopping_client(host="https://sdc-dev.astron.nl:443/", connectors=[ac])

[ ]: # Get data from the shopping basket as a Pandas Dataframe
     basket = shoppingClient.get_basket(convert_to_pandas=True)

[ ]: # Store data as a single Pandas Dataframe
     alta_records = list(basket["alta"]["record"])
     df = pd.DataFrame(alta_records)

[ ]: # Show dataframe
     df

[ ]: # Store data as an astropy table
     from astropy.table import Table
     astropy_table = Table.from_pandas(df)
```

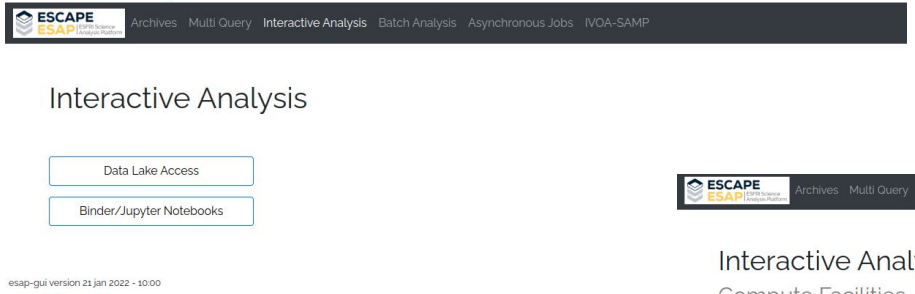

Funded by the European Union's Horizon 2020 - Grant N° 824064

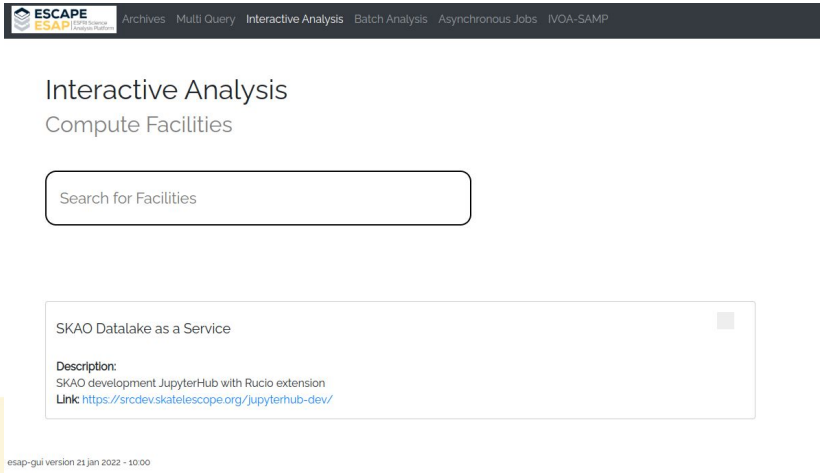
ESAP: Data Lake Access

- The IDA also provides access to a Data Lake as a service
- Developed & available by SKAO
- Implemented via JupyterHub
- Does not require notebook/software selection first

Funded by the European Union's Horizon 2020 - Grant N° 824064

ESAP: Data Lake Access





Funded by the European Union's Horizon 2020 - Grant N° 824064

Future Plans

- Desktop Tools as a Service
  - Deploying a Desktop tool in a container on an IDA environment along with the data
  - Rosetta has been the targets of our initial experiments as a Compute Service that

- Automatic mounting data at IDA facility
  - Moving the data automatically on behalf of the user, so that when deployed, everything (data & software) is available for the user without requiring any additional actions

- ESAP to provision VMs for Interactive Analysis automatically
  - Requires having a compute resources available, and a VM provisioning system (i.e. Openstack) connected and available to ESAP
  - User then can request that an IDA environment become available on a given system, and we can create VM, setup environment, and redirect them to it.

- Associating Software with Data
  - Recommend Software based on what Data has been selected by the user

Funded by the European Union's Horizon 2020 - Grant N° 824064

Thank you!

Funded by the European Union's Horizon 2020 - Grant N° 824064