

# Analyzing gravitational wave data with EOOSC

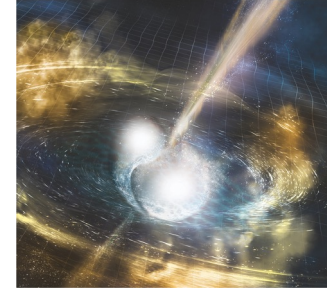
Extreme Universe and Gravitational Waves

A. Iess

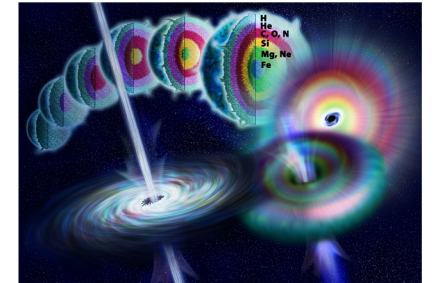
Use cases from the EOOSC Community, 16 May 2023

# The Scientific case in a nutshell

- Detectable Gravitational waves are emitted by extreme events in the Universe involving massive objects (black holes, neutron stars...).
- Provide alerts to other observatories (such as GRB)
- Couple to electromagnetic and neutrino observations in a *multi-messenger framework*



NSF/Ligo/Sonoma State University/ A.Simonnet

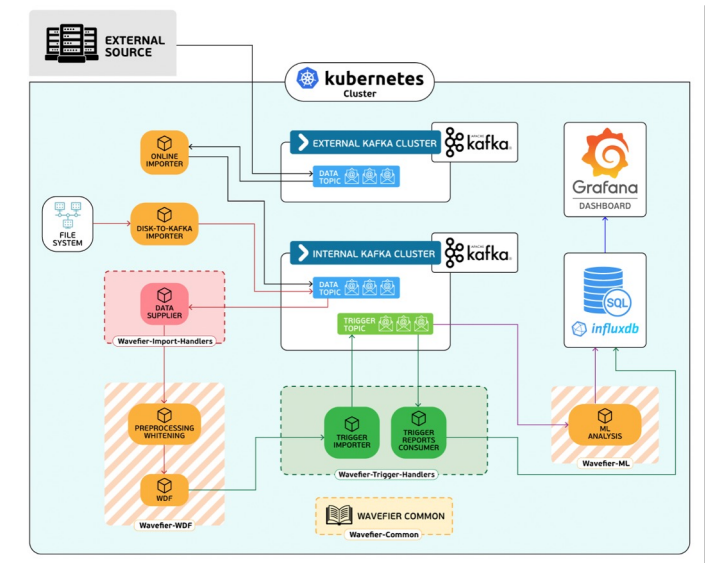


NSF/N.Rager Fuller

In the context of ESCAPE and EOSC Future we are developing a real-time analysis pipeline for multi-messenger data called **Wavefier**.

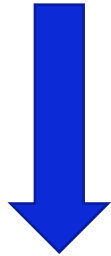
The GW data analysis part is carried out by a specific library integrated in the Wavefier framework, which is the object of this EOSC Community use case, the **Wavelet Detection Filter**.

IN COLLABORATION WITH:  
 **Trust-IT Services**  
communicating to markets



# Leveraging EOSC Future: EOSC Marketplace

For this use case we started from an “agnostic” perspective searching for data on the EOSC Marketplace with simple search queries (“gw”, “gravitational waves”, “LIGO”, “Virgo” etc):



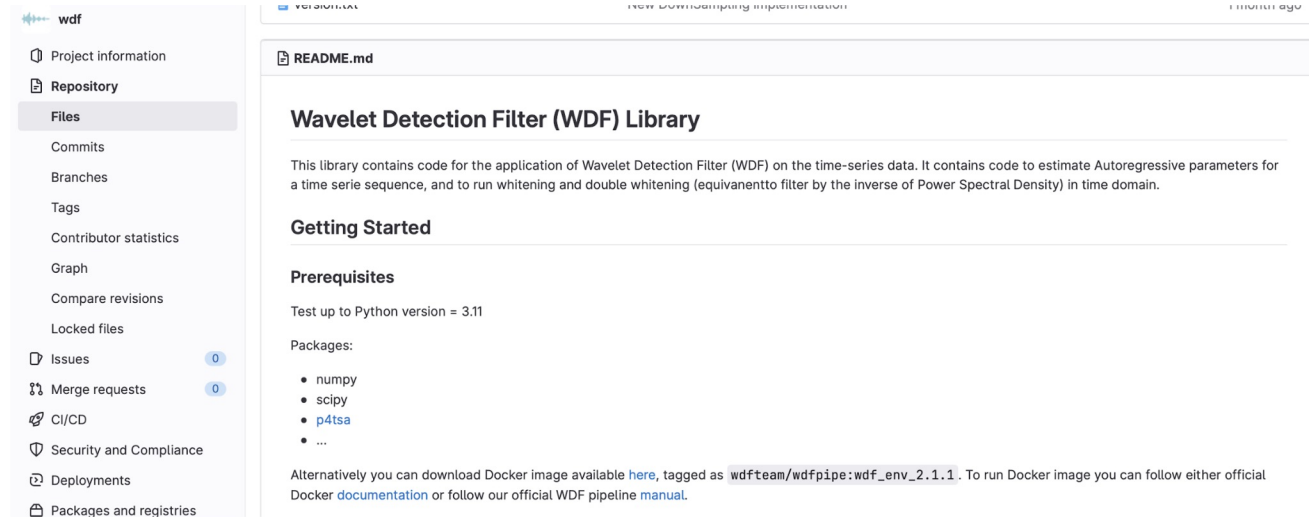
Interesting resource:  
“GWOSC Event Portal Snapshots”

(collection of data characterizing gravitational wave events found during LIGO/Virgo science runs)

The screenshot shows the EOSC Marketplace interface. At the top, it says "Browse EOSC Marketplace Resources" with a search bar containing "gwosc". Below the search bar are navigation icons for "ALL CATALOGS", "PUBLICATIONS", "DATA" (highlighted), "SOFTWARE", "SERVICES", "DATA SOURCES", "TRAININGS", and "OTHER". On the left, there are filter sections for "Research step" (Discover Research Outputs (5)), "Access right" (Open access (5), Restricted (0), Closed (0), Embargo (0)), and "Year range" (Start date). The main content area shows "5 search results Data" with a result for "GWOSC Event Portal Snapshots". This result is a dataset with open access, 44 downloads, and 331 views. The author names are LIGO Scientific Collaboration, Virgo Collaboration, and KAGRA Collaboration. The DOI is 10.5281/zenodo.5602019. The description states that the repository contains snapshots of information available through the GWOSC Event Portal API.

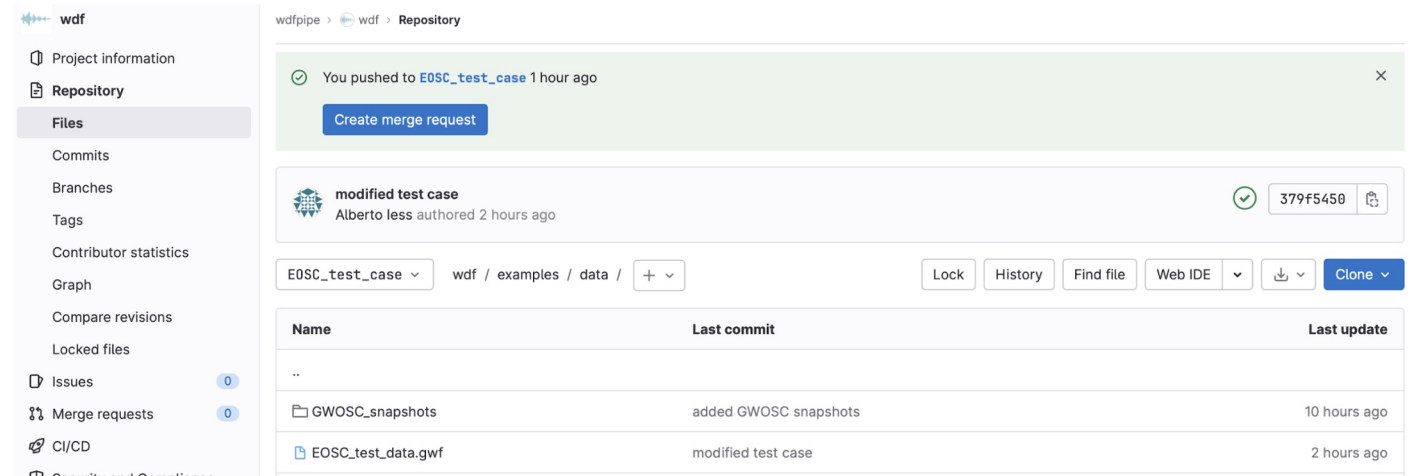
# Gitlab Set Up

- We use a custom branch of the WDF python library, hosted as a gitlab project at:  
[https://gitlab.com/wdfpipe/wdf/-/tree/EOSC\\_test\\_case?ref\\_type=heads](https://gitlab.com/wdfpipe/wdf/-/tree/EOSC_test_case?ref_type=heads)



The screenshot shows the README for the Wavelet Detection Filter (WDF) Library. The page title is "Wavelet Detection Filter (WDF) Library". The content includes a description: "This library contains code for the application of Wavelet Detection Filter (WDF) on the time-series data. It contains code to estimate Autoregressive parameters for a time serie sequence, and to run whitening and double whitening (equivalent to filter by the inverse of Power Spectral Density) in time domain." There is a "Getting Started" section and a "Prerequisites" section. The prerequisites list: "Test up to Python version = 3.11" and "Packages: numpy, scipy, p4tsa, ...". At the bottom, it mentions a Docker image available at [here](#), tagged as `wdfteam/wdfpipe:wdf_env_2.1.1.1`.

- Get the resource found on EOSC Marketplace (indexed in Zenodo) “*GWOSC Event Snapshots*”
- Added to repo for demo purpose.



The screenshot shows the GitLab repository page for the 'wdf' project. A notification at the top says "You pushed to EOSC\_test\_case 1 hour ago" with a "Create merge request" button. Below that, a commit titled "modified test case" by Alberto less is shown, with a commit hash of 379f5450. The breadcrumb navigation is "wdfpipe > wdf > Repository". The file browser shows the path "EOSC\_test\_case > wdf / examples / data /". A table lists the files in the repository:

Name	Last commit	Last update
..		
GWOSC_snapshots	added GWOSC snapshots	10 hours ago
EOSC_test_data.gwf	modified test case	2 hours ago

# Using the EGI-Notebooks

- EGI check-in with academic account
- Obtained membership for specific VO (Virtual Organisations) *vo.notebooks.egi.eu* in order to use EGI Notebooks (~ minutes)

Check-in | Select your Identity

aai.egi.eu/proxy/module.php/discopower/disco.php?entityID=https%3A%2F%2Faai.egi.eu%2Fproxy%2F...

egi  
Check-in

Choose your academic/social account

Search...

- 29 Mays University
- A'SHARQIYAH UNIVERSITY
- A'STAR - Agency for Science, Technology and Research
- A. T. Still University
- AAF Virtual Home
- aai.lab.maeen.sa
- AAI@EduHr Single Sign-On Service
- Aalborg University

or

- Bitbucket
- BitScience
- egi sso
- EGI-IT
- BRACCESS
- Facebook
- GitHub
- Google
- eduTEAMS
- ID OPEN
- IGTF
- LinkedIn
- ORCID
- umbrella
- WeChat
- na

Can't find your identity provider?

English

gnet

Terms Privacy

Copyright ©2016-2022 | Check-in is an EGI service provided by GRNET, receiving funding from the EGI Foundation (EGI.eu) and the EGI-ACE project (Horizon 2020) under Grant number 101017567 | Powered by RCIAM



Turn a Git repo into a collection of interactive notebooks

Have a repository full of Jupyter notebooks? With Binder, open those notebooks in an executable environment, making your code immediately reproducible by anyone, anywhere.

New to Binder? Get started with a [Zero-to-Binder tutorial](#) in Julia, Python, or R.

Build and launch a repository

GitLab.com repository or URL

GitLab.com

Git ref (branch, tag, or commit) Path to a notebook file (optional)

Copy the URL below and share your Binder with others:

Expand to see the text below, paste it into your README to show a binder badge:

```
[[Binder]](https://replay.notebooks.egi.eu/badge_logo.svg) (https://replay.notebooks.egi.eu/v2/gl/wdfpipe%2Fwdf/EOSC_test_case)
```



# Performing The Analysis

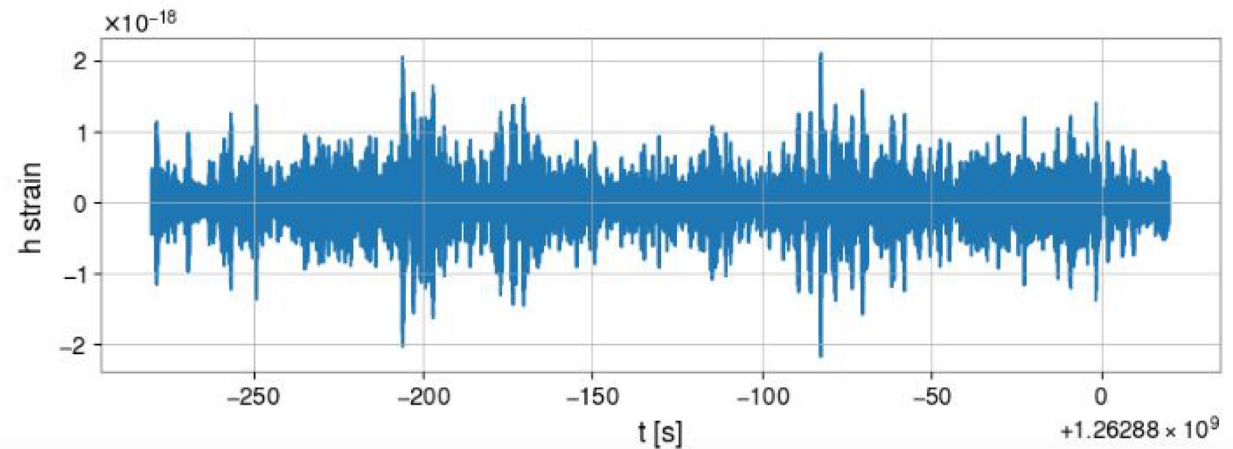
Notebook to run analysis starting from the GWOSC Event Snapshots (using GPS times)

- Binder to easily reproduce environment and notebook through github, gitlab and other repositories.

```
In [94]: 1 event['events']['GW200112_155838-v1']['strain'][0] #['GPSstart']
Out[94]: {'GPSstart': 1262879920,
'detector': 'L1',
'duration': 32,
'format': 'gwf',
'sampling_rate': 16384,
'url': 'https://www.gw-openscience.org/eventapi/json/GWTC-3-confident/GW200112_155838/v1/L-L1_GWOSC_16KHZ_R1-1262879920-32.gwf'}
```

We use the data stored in the event snapshot to fetch **open data** from the GWOSC platform (using gwpy installed in the environment, details at <https://gwpy.github.io/docs/stable/timeseries/opendata/>)

```
In [198]: 1 gps_event = event['events']['GW200112_155838-v1']['strain'][12]['GPSstart']
2 start = int(gps_event) - 200
3 end = int(gps_event) + 100
4 data_L1 = TimeSeries.fetch_open_data('L1', start, end)
5 plt.figure(figsize=(10,3))
6 plt.plot(data_L1, plt.xlabel('t [s]'), plt.ylabel('h strain');
```

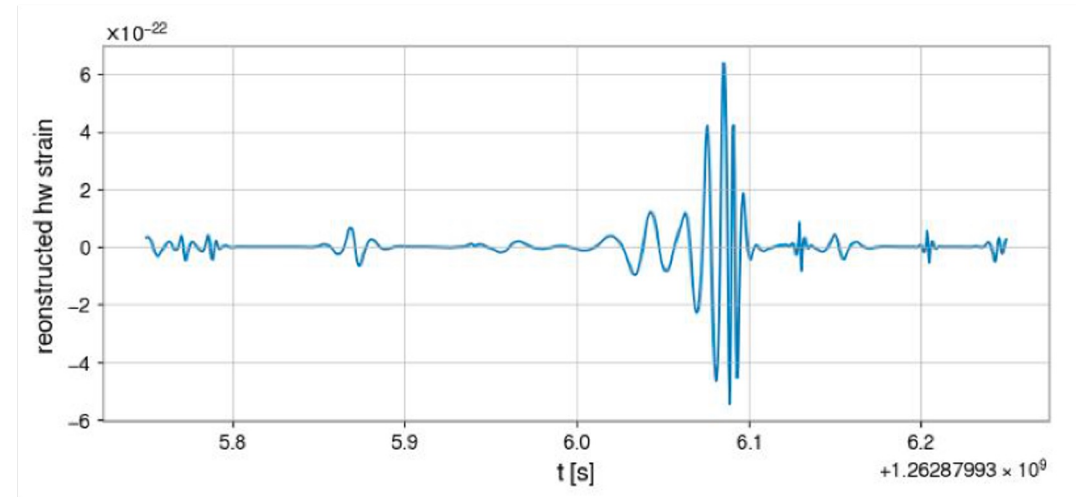
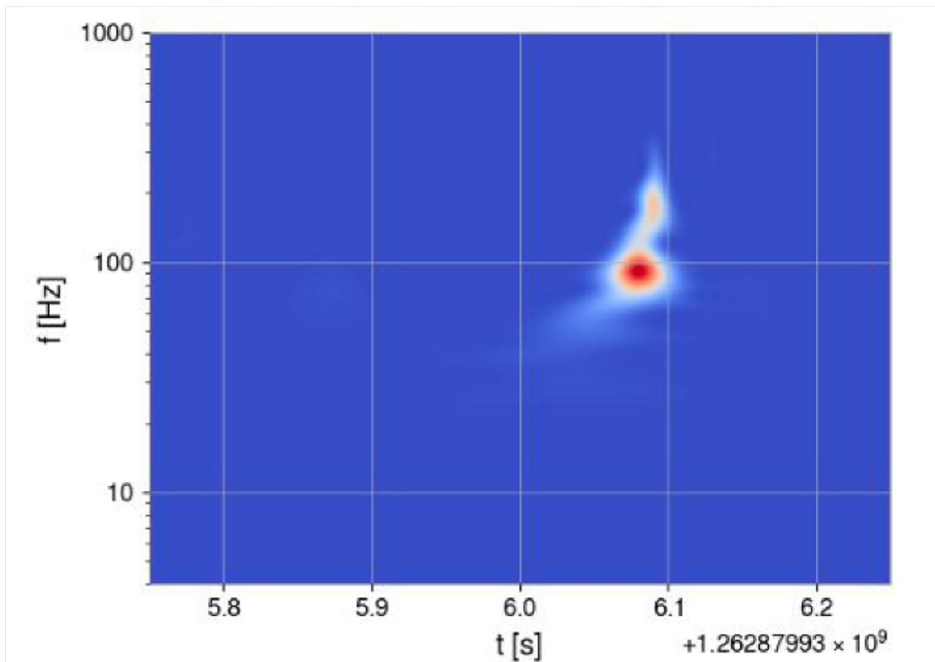


# Performing The Analysis

- Transient event list in .csv file
- Reconstructed whitened chirp signal as a time series and in the time-frequency domain


```
In [145]: 1 wdf=wdfUnitDSWorker(par,fullPrint=2)
          2 wdf.segmentProcess(par.segment)
2023-05-15 20:14:52 hal2022 root[20277] INFO Analyzing segment: 1262879720-1262880020 for channel L1:GWOSC-16KHZ_
R1 STRAIN downsampled at 2048Hz
2023-05-15 20:14:52 hal2022 root[20277] INFO Start AR parameter estimation
2023-05-15 20:15:47 hal2022 root[20277] INFO Estimated sigma= 2.284710935695034e-22
2023-05-15 20:16:14 hal2022 root[20277] INFO Starting detection loop
2023-05-15 20:23:08 hal2022 root[20277] INFO analyzed 300 seconds in 495.7170033454895 seconds
```

	gps	gpsPeak	duration	EnWDF	snrMean	snrPeak	freqMin	freqMean	freqMax	freqPeak	...	rw1014	rw1015	rw1016	rw1017
0	1262879725.125	1262879725.447	0.500	0.206	0.160	0.794	46.000	105.500	200.000	72.000	...	0.000	0.000	0.000	0.000
1	1262879727.625	1262879727.749	0.500	0.208	0.180	1.380	46.000	150.692	340.000	140.000	...	0.000	-0.000	-0.000	-0.000
2	1262879727.875	1262879728.122	0.493	0.203	0.161	1.033	44.000	121.000	248.000	56.000	...	0.000	-0.000	-0.000	0.000
3	1262879735.375	1262879735.800	0.488	0.236	0.197	1.509	76.000	169.385	258.000	152.000	...	-0.000	-0.000	-0.000	-0.000
4	1262879735.500	1262879735.800	0.298	0.223	0.183	1.509	98.000	170.577	252.000	128.000	...	0.000	0.000	0.000	0.000
5	1262879735.625	1262879735.800	0.411	0.214	0.179	1.502	98.000	170.654	252.000	128.000	...	0.000	-0.000	-0.000	-0.000
6	1262879735.750	1262879735.800	0.315	0.211	0.165	1.512	98.000	174.346	272.000	158.000	...	-0.000	-0.000	0.000	0.000
7	1262879739.625	1262879739.903	0.492	0.225	0.180	1.101	54.000	141.000	274.000	72.000	...	0.000	0.000	0.000	0.000
8	1262879739.750	1262879739.903	0.472	0.247	0.199	1.108	54.000	139.615	260.000	126.000	...	-0.000	-0.000	-0.000	-0.000
9	1262879739.875	1262879739.903	0.480	0.293	0.222	1.155	54.000	140.192	280.000	130.000	...	-0.000	-0.000	-0.000	-0.000





# Use case story in a nutshell

- Used EOSC Marketplace to find a resource indexed in the EOSC portal, through Zenodo.
- Downloaded resource to find astrophysical event GPS time.
- Tested accessing EGI resources through identity check
- Tested EGI notebook to reproduce analysis.
- Used resource to download existing open source data from the Gravitational Wave Open Science Center.
- Carried-out simple gravitational wave data analysis using Wavelet Detection Filter (gitlab, soon on Zenodo  visible from the Marketplace)

## Added value by EOSC

- Data found through Marketplace
- Computing resources EGI-Notebook
- Reproducibility through Binder
- Student Training