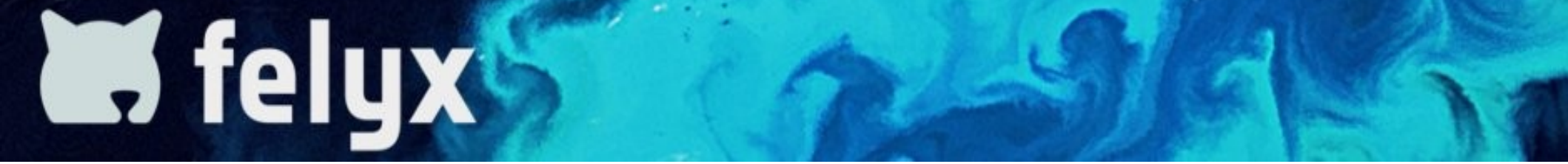


Felyx : application to GHRSST activities

Jean-François Piollé – **Ifremer**

Sylvain Herlédan – **OceanDataLab**

Igor Tomazic – **Eumetsat**

Philippe Goryl, Craig Donlon, Veronica Guidetti – **ESA**

esa

Ifremer

pelamis scientific software

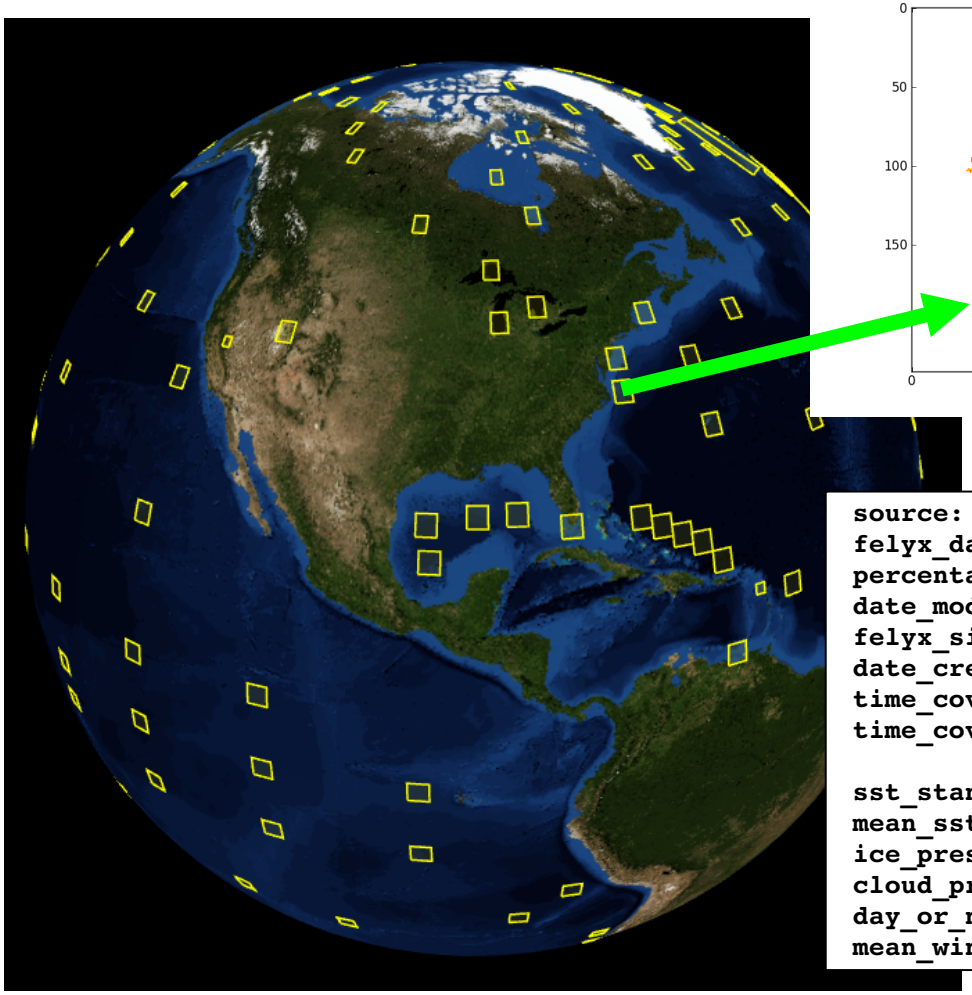
PML Plymouth Marine Laboratory

felyx

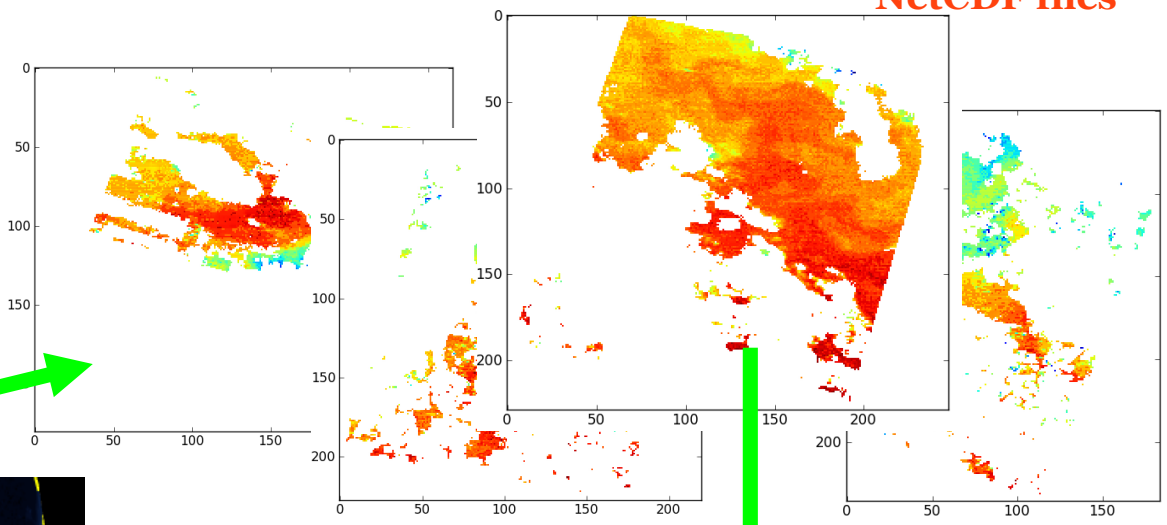
back-end concept

extract **miniprods** (subsets) over static and dynamic sites

process quantitative, qualitative, stat metrics over miniprods



NetCDF files




```
source: 20130101-IFR-L4_GHRSST-SSTfnd-ODYSSEA-GL_B_010-v2.0-fv1.0.nc
felyx_dataset_name: ifr-l4-sstfnd-odyssea-glob_0_0_v2.1
percentage_coverage_of_site_by_miniprod: 100.0
date_modified: 2014-04-18T10:30:21
felyx_site_identifier: ukm005
date_created: 2014-04-18T10:30:21
time_coverage_start: 2013-01-01T00:00:00
time_coverage_stop: 2013-0101T00:00:00

sst_standard_deviation : 1.34
mean_sst : 286.289
ice_presence: 0
cloud_presence": 46.80
day_or_night: "night"
mean_wind_speed: 4.8388
```

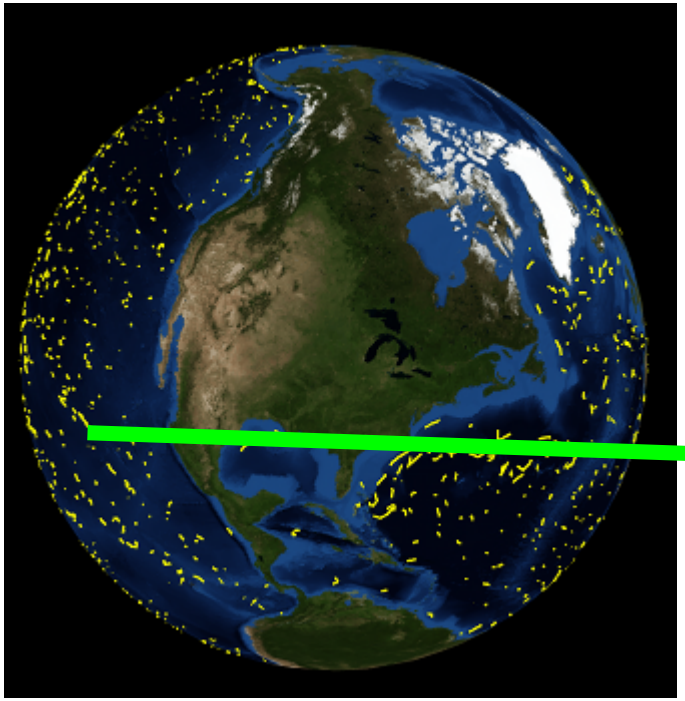

JSON files
indexed in a search
engine (ElasticSearch)

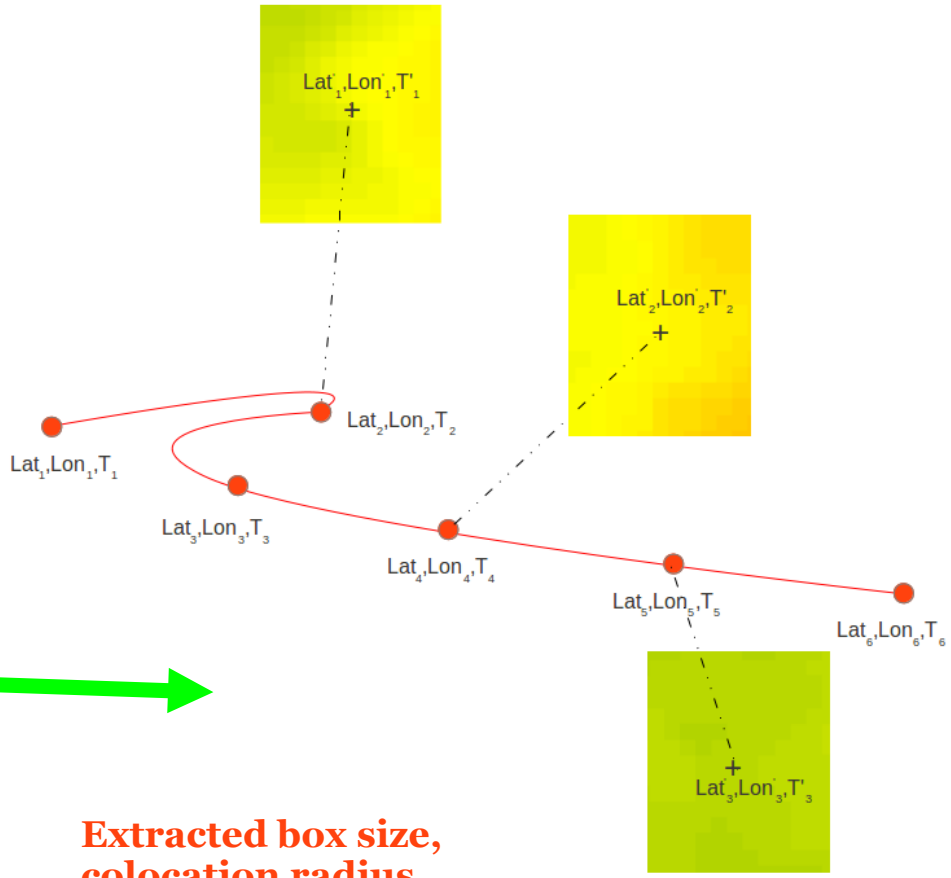
sites may be trajectories (buoys, cruise, hurricane)

sites may be trajectories (buoys, cruise, hurricane)

MINIPROD's centred on trajectory locations closest in time locations closest in time

trajectory files ingested through import web service (CSV file)

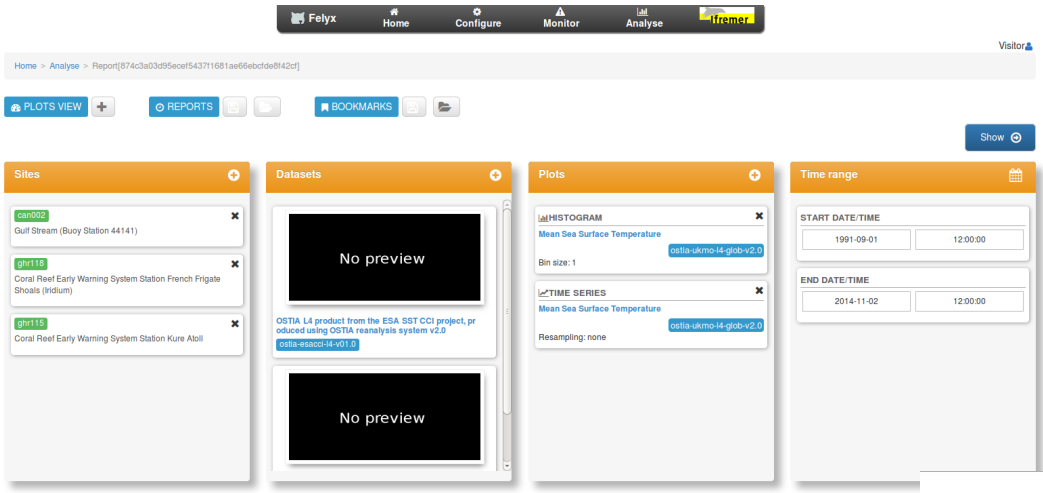


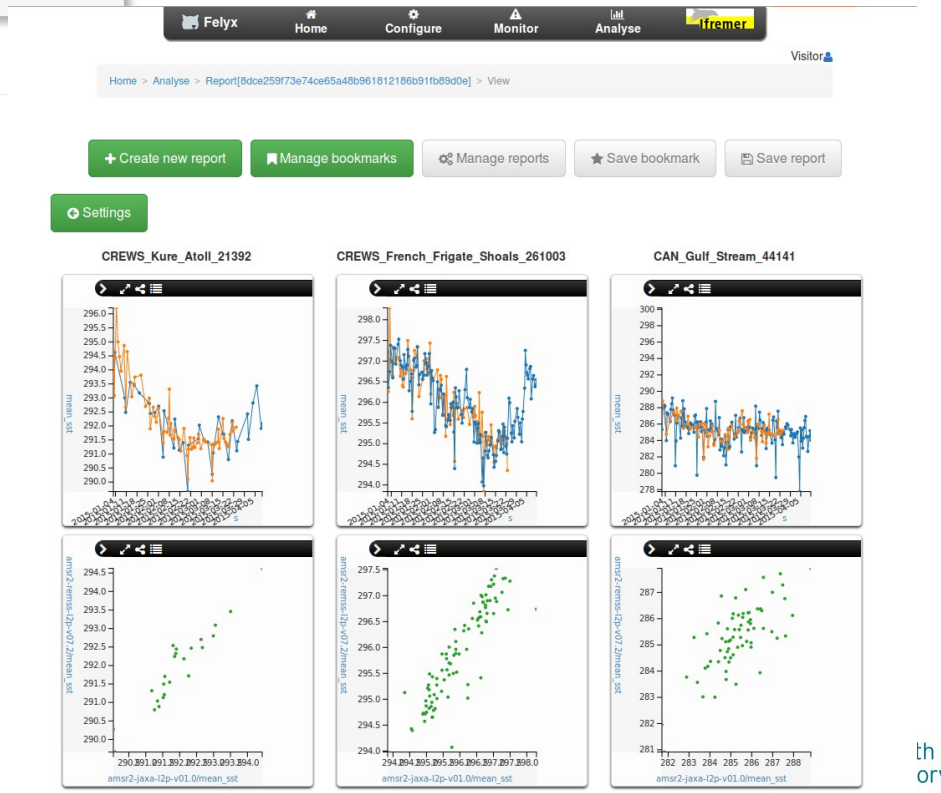


Extracted box size, colocation radius, maximum temporal difference can be adjusted for each dataset

felyx

Web reporting interface





Funded by the European Space Agency
Designed and built by Ifremer with the support of Pelamis and PML.
Code licensed under GPLv3, documentation under Sphinx.

esa Ifremer PML Plymouth Marine Laboratory pelamis

Contact administrator

esa

th Marine
ory

felyx

Access and visualization

In [4]:

```
from datetime import datetime

from pyfelyx.query.selection import MetricSelection
from pyfelyx.query.instance import Instance

inst = Instance(url='http://localhost:1080/felyx')

query = MetricSelection(
                            sites=['cci53057'],
                            datasets=['ostia-esacci-l4-v01.0'],
                            metrics=['matchup_sst', 'insitu_sst'],
                            start=datetime(2001,1,1),
                            end=datetime(2001,12,31)
                            )

result = query.run(inst)
```

In [12]:

```
from datetime import datetime

from pyfelyx.query.selection import MetricSelect
from pyfelyx.query.instance import Instance

inst = Instance(url='http://localhost:1080/felyx

query = MetricSelection(
                            sites=['cci53057'],
                            datasets=['ostia-esa
                            metrics=['matchup_ss
                            start=datetime(1991,
                            end=datetime(2010,12
                            )

result = query.run(inst)
```

In [5]:

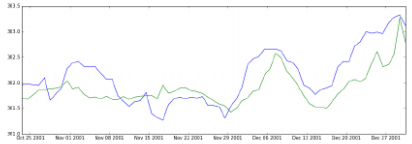
```
matchup_sst = result['results']['ostia-esacci-l4-v01.0']['cci53057']['metr
ics']['matchup_sst']
insitu_sst = result['results']['ostia-esacci-l4-v01.0']['cci53057']['metri
cs']['insitu_sst']
times = result['results']['ostia-esacci-l4-v01.0']['cci53057']['times']
```

In [13]:

```
matchup_sst = result['results']['ostia-esacci-l4
ics']['matchup_sst']
times = result['results']['ostia-esacci-l4-v01.0
```

In [6]:

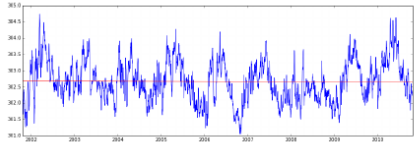
```
%matplotlib inline
import matplotlib.pyplot as plt

fig = plt.figure(figsize=(15,5))

plt.plot(times, matchup_sst)
plt.plot(times, insitu_sst)
plt.show()
```



In [14]:

```
%matplotlib inline
import matplotlib.pyplot as plt

fig = plt.figure(figsize=(15,5))

plt.plot(times, matchup_sst)

#Linear regressison -polyfit
from scipy import polyval, polyfit, sqrt

days = [(_- datetime(2007,1,1)).days for _ in times]
(ar,br)=polyfit(days,matchup_sst,1)
xr=polyval([ar,br], days)

print('Linear regression using polyfit')
print('regression: a=%f b=%f' % (ar,br))
plt.plot(times,xr,'r-')

plt.show()
```

```
Linear regression using polyfit
regression: a=-0.000012 b=302.664413
```



API

Allow to query, process and display programmatically metrics and miniprods

Build custom diagnostics and plots

Cross-queries

Example : usage with iPython notebooks

FTP/OpenDAP access to miniprods can be offered too.

esa

Ifremer

pelamis scientific software

PML Plymouth Marine Laboratory

Status

being used and tested in different contexts : bug detection, improvements and new functionalities being added to match the selected applications

Miniprod extraction fully functional

Full multi-sensor match-up creation with additional in situ data server being tested

Simplified API being finalized

Full release of the software to anybody is planned once these applications have been fully validated

User case 1 : SMOS Storm

Objective : better estimation of strong winds using synergy between SMOS and other sources.

Tasks : build a catalogue of data subsets from various sources of observation, ordered by storm

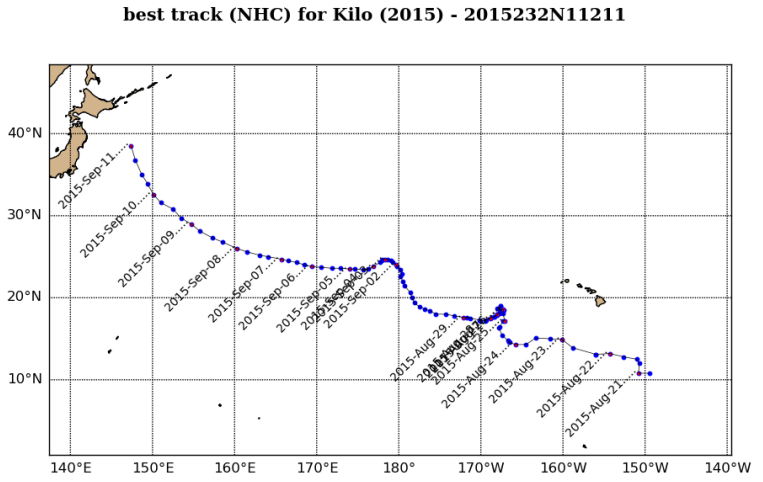
How felyx helped : perform colocation with storm tracks and extract data subset along this track at storm time (+/- 3 hours) – used as an extraction tool in common format (NetCDF CF)

Inputs

IBTRacS, NHC,... for hurricane trajectories

EO data :

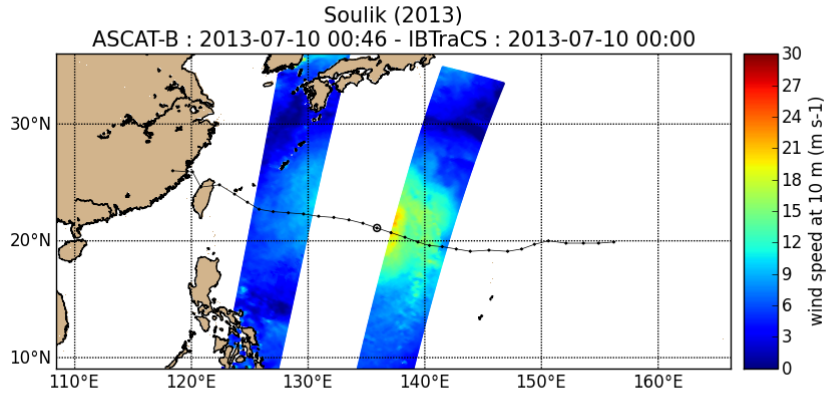
- SMOS, AMSR-E & SMAP (ESA/SMOSStorm),
- Jason-1/Jason-2/AltiKa/Cryosat-2 data from GlobWave
- ASCAT A & B 12.5 km data from OSI SAF, *TBD* : OSCAT, HY2A, RapidScat
- Windsat
- Model and blended data : Hwind,...

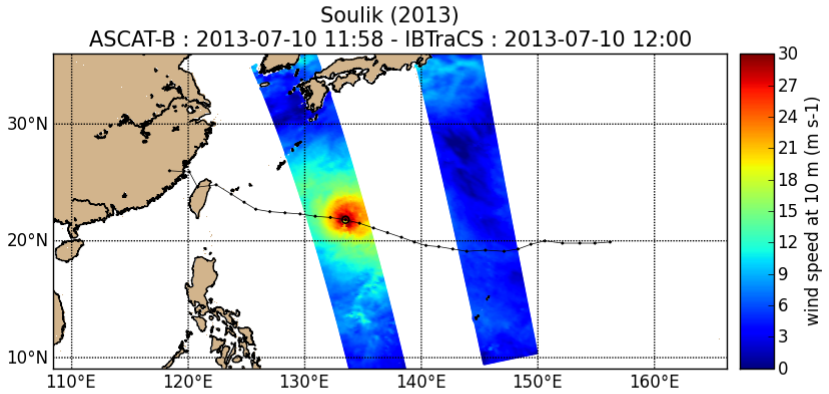


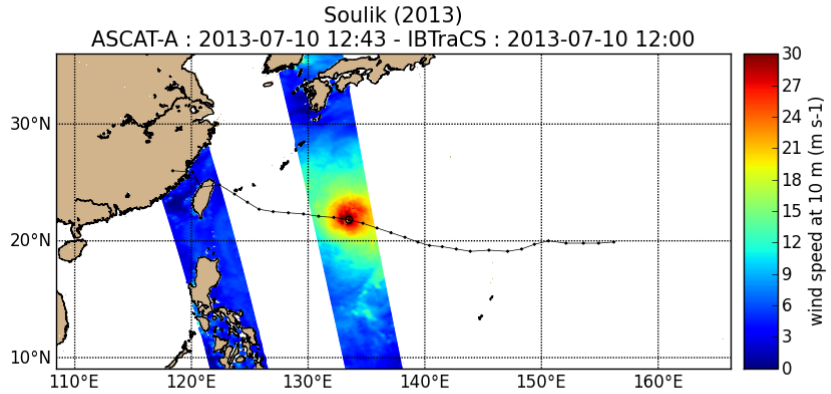
http://www.smosstorm.org

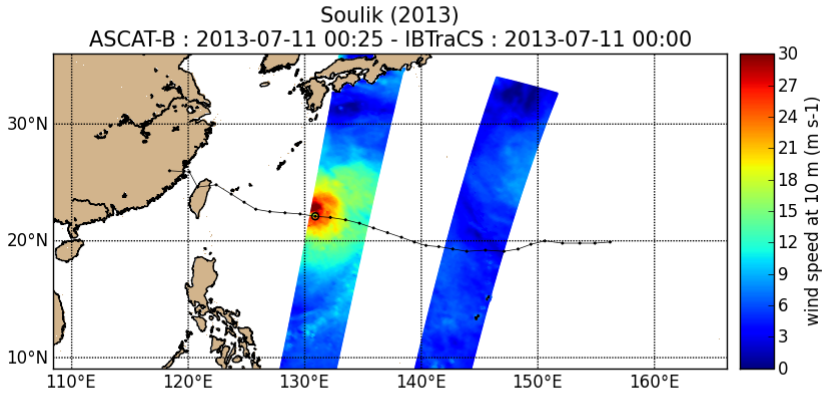
Ifremer pelamis scientific software PML Plymouth Marine Laboratory

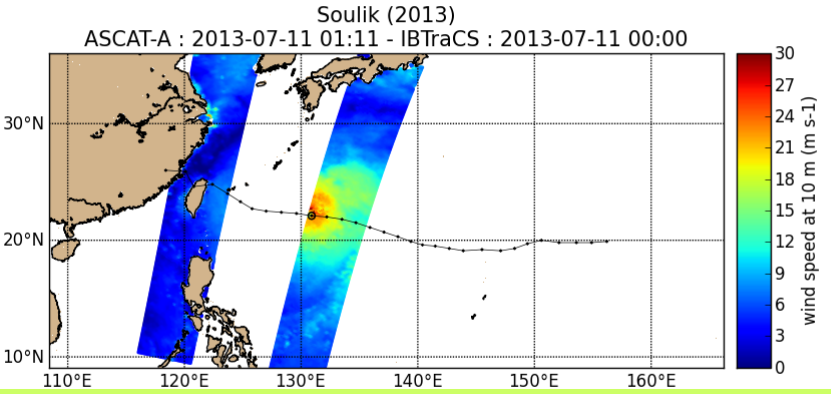
User case 1 : SMOS Storm

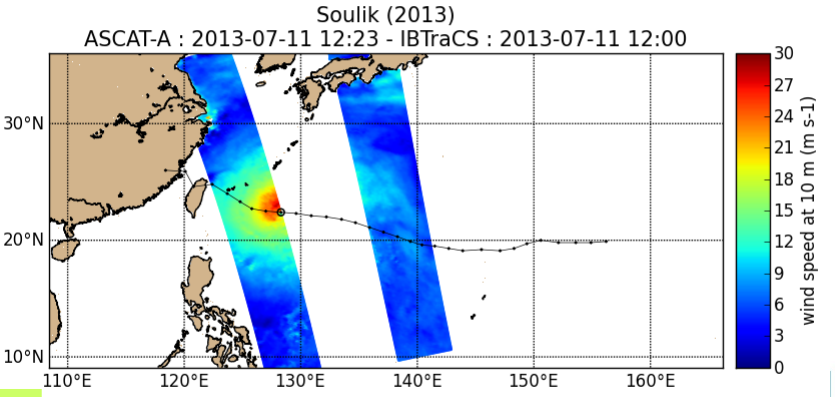












http://www.ifremer.fr/cersat/images/smosstorm2

Plymouth Marine Laboratory

Use case 2 : Sentinel-3 MDB

Context : cal/val Sentinel-3 / SLSTR – specific project supported by Eumetsat through OSI SAF FA to build a multi-sensor match-up database

Requirements for match-ups defined by a dedicated team, implementation and operation centralized at Ifremer using felyx framework

Coordination with Copernicus/CMES (Coriolis) for delivery of in situ data following cal/val community requirements (workshop Ifremer, January 2016)

Sentinel-3 MDB : workflow

Colocation window : 2h, 5km

21 x 21 pixel boxes

± 6h of in situ data history

In situ data :

- Copernicus/CMEMS (Coriolis)
- ISAR radiometer on opportunity ships (delayed-mode)

Sentinel-3 data :

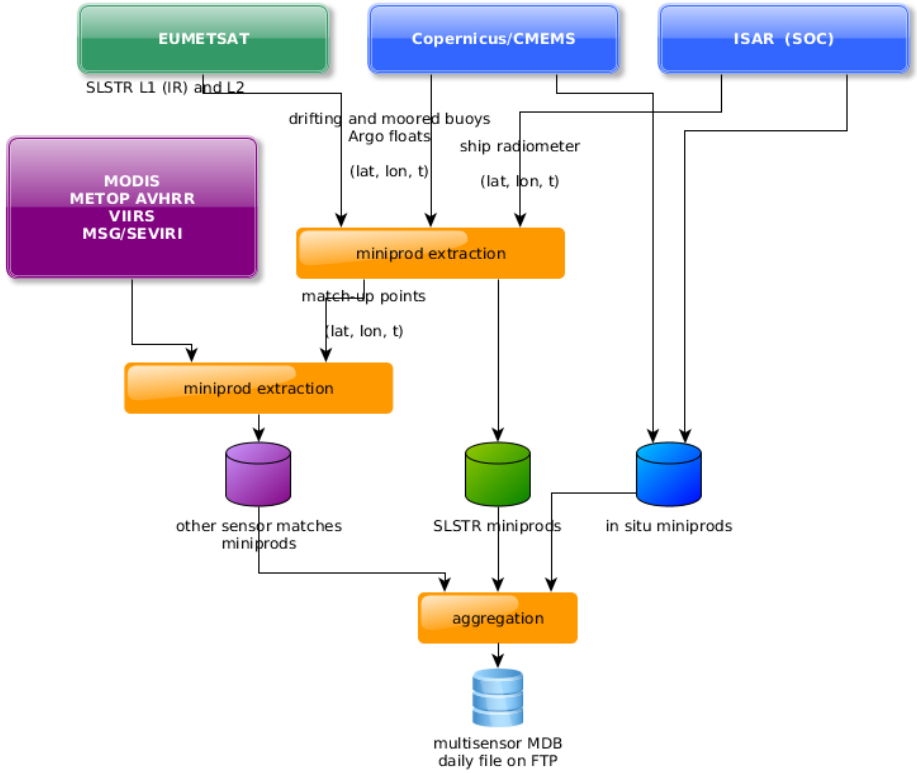
- L1 infra-red channels
- L2 (SST) – all fields, incl. meteo and ancillary fields

Other sensor data

- Metop-B/AVHRR, MODIS, VIIRS, MSG/SEVIRI

Resampling of all data to SLSTR grid

Daily aggregated match-up files on FTP : stack all matchups into a single file.



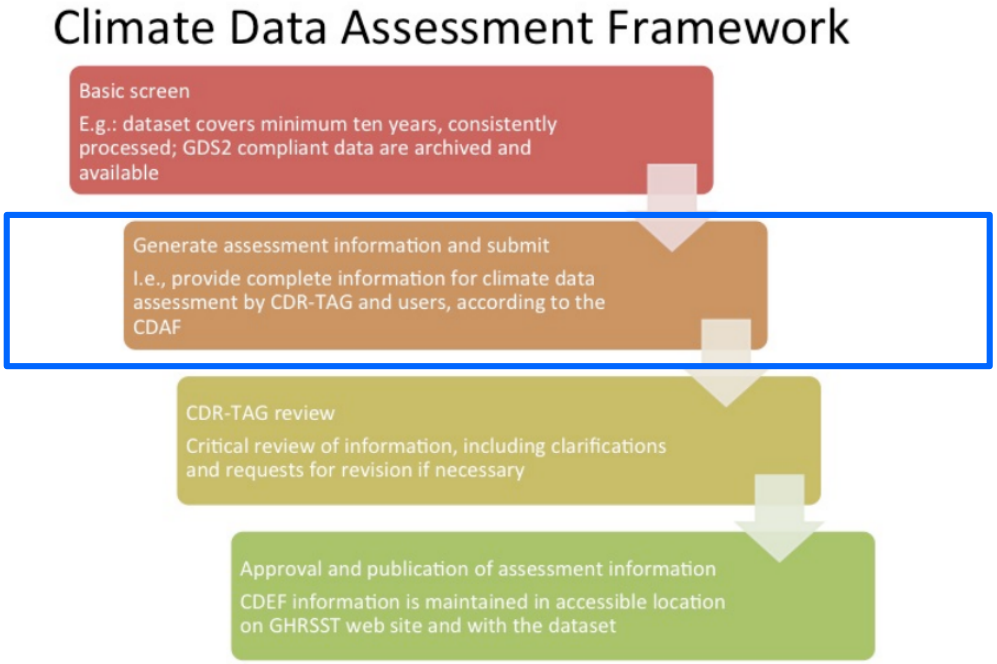
Use case 1 : essential climate variables

Context : GHRSST Climate Data Assessment Framework (CDAF) - « Creating a community multi-sensor match-up system is a GHRSST objective »

How to assess if a dataset is suitable for climate trend detection ?

Accuracy, sensitivity and consistency

Provide a tool to users to evaluate this on their own dataset using a common set of diagnostics and reference data



More resources

Documentation : felyx.readthedocs.org

Source code and packages : python implementation, git server, GPLv3 open source licence. Access link will be provided on http://felyx.org

Virtual Machine (virtual box) available for testing.

Useful dependencies :

Cerbere package : generic API to various formats and observation patterns.

Documentation : http://cerbere.readthedocs.fr

Git : https://git.cersat.fr/cerbere

Contact : jfpiolle@ifremer.fr