

Python tools for the JT60-SA data access

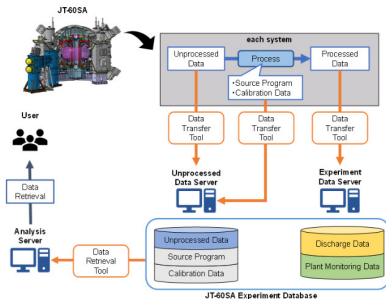
Matteo Iafrati

EUROfusion - ENEA

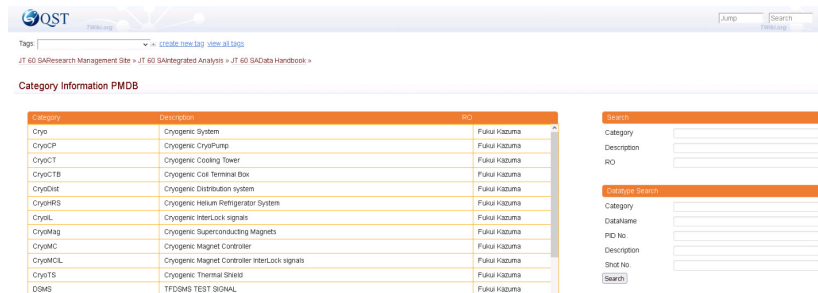
2022 Sep 06

- ▶ JT60-SA Database System
- ▶ Tools for data access
- ▶ Conclusion and future work

The JT-60SA experiment data are categorized into three types: (i) discharge data, (ii) plant monitoring data, and (iii) unprocessed data. We will focus on the Experimental Data DataBase (EDDB) and the Plant Monitor DataBase (PMDB)



All the information are on the QST wiki



QST
7966.org

Jump Search
7966.org

Tags: [create new tag](#) [view all tags](#)

[JT 60 SAR Research Management Site](#) » [JT 60 SA Integrated Analysis](#) » [JT 60 SA Data Handbook](#) »

Category Information PMDB

Category	Description	RO
Cryo	Cryogenic System	Fukui Kazuma
CryoCP	Cryogenic CryoPump	Fukui Kazuma
CryoCT	Cryogenic Cooling Tower	Fukui Kazuma
CryoCTB	Cryogenic Coil Terminal Box	Fukui Kazuma
CryoDist	Cryogenic Distribution system	Fukui Kazuma
CryoHRS	Cryogenic Helium Refrigerator System	Fukui Kazuma
CryoIL	Cryogenic InterLock signals	Fukui Kazuma
CryoMag	Cryogenic Superconducting Magnets	Fukui Kazuma
CryoMC	Cryogenic Magnet Controller	Fukui Kazuma
CryoMCL	Cryogenic Magnet Controller InterLock signals	Fukui Kazuma
CryoTS	Cryogenic Thermal Shield	Fukui Kazuma
DSMS	TFDSMS TEST SIGNAL	Fukui Kazuma

Search

Category

Description

RO

Datatype Search

Category

DataName

PID No.

Description

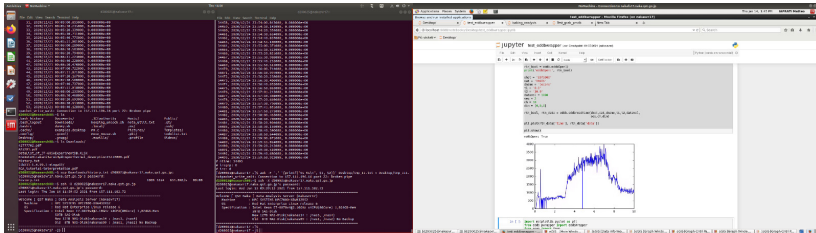
Shot No.

How to access the data?

- ▶ *"eGIS is a software to visualize the time series experimental data during discharge"*
- ▶ Web API on JT-60SA Experiment Database
- ▶ Discharge Data Access Libraries and Plant Data Access Libraries (C and Python)

Preliminary work on the databases

Data access using libraries or .exe executable



Access to the RGA data and download using EGIS

In [6]: `!ls ../edas/egis/work/`

```
1st_baking_bis.egis_ws core.64100
1st_baking_bis.mplset egis_user_nakasvr17.ini
1st_baking.csv test_workspace_GLW.egis_ws
1st_baking.egis_ws test_workspace_GLW.mplset
1st_baking.mplset test_workspace_QMS_2.egis_ws
2nd_baking.csv test_workspace_QMS_2.mplset
2nd_baking.egis_ws test_workspace_QMS.egis_ws
2nd_baking.mplset test_workspace_QMS.mplset
core.15795 WV_port_temp.egis_ws
core.50457 WV_port_temp.mplset
core.57369
```

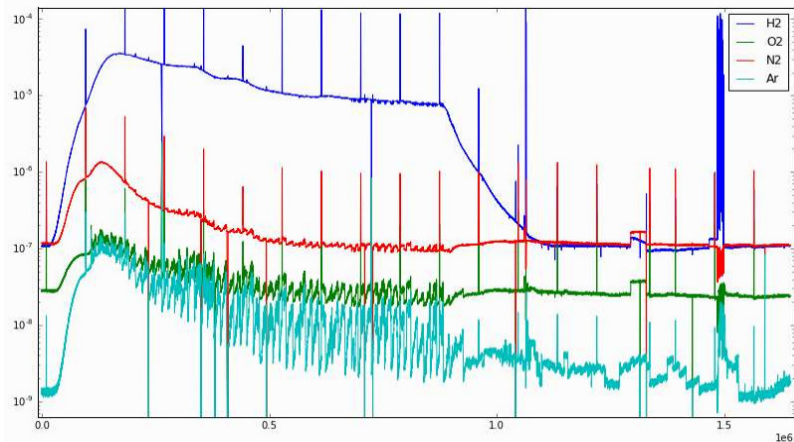
In [337]: `import pandas as pd`

```
data = pd.read_csv(r'../edas/egis/work/1st_baking.csv', skiprows = 1, low_memory=False)
#df = pd.read_csv('../edas/egis/work/1st_baking.csv')
#df = pd.DataFrame(data, columns= ['/QMS/preMidC', '/QMS/mz02c'])
#print (df)
pd.options.display.max_columns = 100
data.head(3)
```

Out[337]:

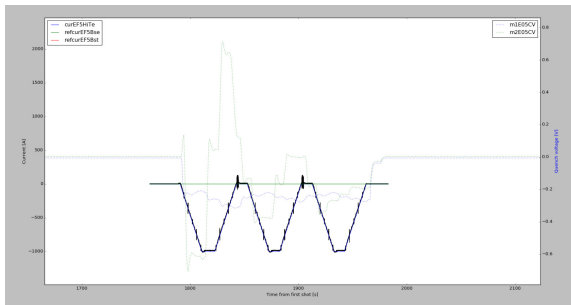
	# DATA-ID d01	d01	d04	d04.1	d05	d05.1	d07	d07.1	d08
0	# shotnum	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	# dataname /QMS/preMidC	/QMS/preMidC	/QMS/mz01c	/QMS/mz01c	/QMS/mz02c	/QMS/mz02c	/QMS/mz14c	/QMS/mz14c	/QMS/mz16c
2	2020/11/29 10:00:00.000000	3.40000e-05	2020/11/29 10:00:00.125000	5.14000e-08	2020/11/29 10:00:00.125000	1.06000e-07	2020/11/29 10:00:00.125000	8.82000e-09	2020/11/29 10:00:00.125000

RGA data with correction factor



EDDB and PMDB data plotted together

eddb_pywrapper library and the plantmonitor.exe have been used in order to retrieve data and plot them using the same time axis.



EDDB and PMDB data access example

```
"""
#author M.I.

import os
import sys
import numpy as np
import matplotlib.pyplot as plt
import datetime
from statistics import mean

def unix_time_millis(dt, epoch):
    return (dt - epoch).total_seconds()

def date_convert(date to convert):
    return datetime.datetime.strptime(date_to_convert, '%Y/%m/%d %H:%M:%S.%f')

def date_convert_II(date to convert):
    return datetime.datetime.strptime(date_to_convert, '%Y%m%d%H%M%S.%f')

from ctypes import *
def import_file(full_path_to_module):
    try:
        import os
        module_dir, module_file = os.path.split(full_path_to module)
        module_name, module_ext = os.path.splitext(module_file)
        save_cwd = os.getcwd()
        os.chdir(module_dir)
        module_obj = import (module_name)
        module_obj._file_ = full_path_to module
        globals()[module_name] = module_obj
        os.chdir(save_cwd)
    except Exception as e:
        raise ImportError(e)
    return module_obj

from enum import Enum

class datatype(Enum):
    SHORT = 0
    INT = 1
    FLOAT = 2
    DOUBLE = 3

PATH_TO_LIB = '/analysis/lib/libeddb.so'
lib = cdll.LoadLibrary(PATH_TO_LIB)
import_file('/analysis/src/eddb/eddb_pwrapper.py')

from eddb_pwrapper import eddbwrapper

eddb = eddbwrapper(PATH_TO_LIB)
```

```
124
125 tmp_string = "bash "+os.path.abspath("tmp.sh")
126
127 for i in range(len(pmdb_CAT)):
128     cmd_string = "plantload.exe "+pmdb_CAT[i]+" "+pmdb_dname[i]+" "+start+" "+end+" > tmp.txt"
129     os.system(cmd_string)
130     os.system(tmp_string)
131     #os.system("./tmp.sh")
132     pmdb_data = np.loadtxt('tmp.txt', dtype='bytes').astype(str)
133     pmdb_time_list = []
```

Different approaches to access the two databases:

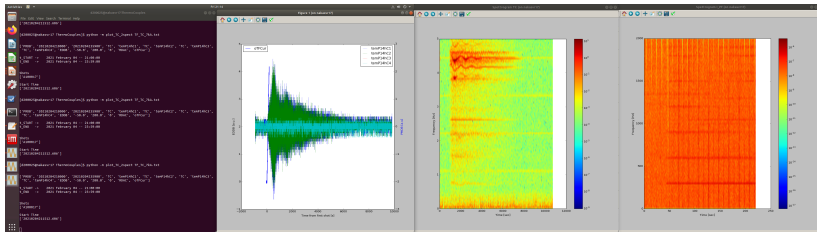
- ▶ using EDDB library
- ▶ calling an external .exe executable

During 2021 and 2022 many updates have been released:

- ▶ EGIS now support to access and plot signals from both databases
- ▶ the eddb_pywrapper library (C or Python) is much easy to import in scripts now
- ▶ the new pmdb_pywrapper library (C or Python) is now available

More sophisticated analysis

Once you have a quick data access you can perform data analysis using the tools you are more confident





"Web API on JT-60SA Experiment Database is an interface for downloading data from JT-60SA experiment database system via a network."

Web API on JT-60SA Experiment Database

? [Japanese](#) [English](#)

[Experiment](#) [Plant](#)

pid ✓

rdata Binary Info ASCII

t1 ✓

t2 ✓

out

URL

Not Found

The time range specified by the "t1" and "t2" arguments is over 86400 sec.

- ▶ Quick data access is possible using EGIS, WebAPI or libraries (C or Python)
- ▶ Many updates during the last year
- ▶ More users could help the developing and Continuous Integration process.