Fusion Data Framework Documentation

Release 0.0.0

John Schmitt, David R. Smith, Kevin Tritz, Howard Yuh

October 06, 2015

CONTENTS

CHAPTER

ONE

ABOUT FDF

Fusion Data Framework (FDF) is a data access, management, and visualization framework for magnetic fusion experiments.

Repository: https://github.com/Fusion-Data-Framework/fdf

Documentation: HTML or PDF

Submit bugs or feature requests: https://github.com/Fusion-Data-Framework/fdf/issues

Created by:

- John Schmitt, Princeton Plasma Physics Lab
- David R. Smith, U. Wisconsin-Madison
- Kevin Tritz, The Johns Hopkins U.
- Howard Yuh, Nova Photonics

To contribute to the FDF project, please contact John, David, Kevin, or Howard.

CHAPTER

TWO

GETTING STARTED GUIDE

2.1 User Guide

2.2 Developer Guide

CHAPTER

THREE

USAGE EXAMPLES

CHAPTER

FOUR

PACKAGE REFERENCE

4.1 Module factory.py

Root module for the FDF package.

Classes

- Machine - root class for the FDF package
- Shot - shot container class
- Logbook - logbook connection class
- Container - diagnostic container class
- Node - mdsplus signal node class
- FdfError - error class for FDF package

Usage:

```
>>> import fdf
>>> nstx = fdf.Machine('nstx')
>>> nstx.s140000.logbook()
>>> nstx.addshots(xp=1048)
>>> nstx.s140000.mpts.plot()
```

4.2 Class Machine

class `factory.`**`Machine`**(*name='nstx'*, *shotlist=[]*, *xp=[]*, *date=[]*)

Factory root class that contains shot objects and MDS access methods.

Basic class initialization is performed as follows: >>>nstx = Machine(name='nstx')

the Machine class contains a model shot object: nstx.s0

shot data can be accessed directly through the Machine class: >>> nstx.s141398 >>> nstx.s141399

alternatively, a list of shot #'s may be provided during initialization: >>>nstx = Machine(name='nstx', shotlist=[141398, 141399])

or added later using the addshot method: >>>nstx.addshot([141398, 141399])

4.3 Class Shot

class `factory.`**`Shot`**(*shot*, *root=None*, *parent=None*)

4.4 Class Logbook

class `factory.`**`Logbook`**(*name='nstx'*, *root=None*)

4.5 Module fdf_signal.py

fdf-signals.py - module containing Signal class

Classes

- Signal - signal class for data objects

Created on Tue Jun 23 2015

@author: hyuh

4.6 Class Signal

class `fdf_signal.`**`Signal`**(***kwargs*)

sig=fdf.Signal(signal_ndarray, units='m/s', axes=['radius','time'], axes_values=[ax1_1Darray, ax2_1Darray], axes_units=['s','cm']

e.g.: mds.Signal(np.arange((20*10)).reshape((10,20)), units='keV', axes=['radius','time'], axes_values=[100+np.arange(10)*5, np.arange(20)*0.1], axes_units=['s','cm'])

or an empty signal: s=mds.Signal() default axes order=[time, space] sig=fdf.Signal(units='m/s', axes=['radius','time'], axes_values=[radiusSignal, timeSignal])

4.7 Module fdf_globals.py

fdf_globals.py contains package-level constants

Created on Thu Jun 18 11:18:16 2015

@author: ktritz

CHAPTER

FIVE

LICENSE

CHAPTER

SIX

INDICES AND TABLES

- genindex
- modindex
- search

f

F

L

M

S