



Refreshment tutorial on environment, FC2K and kepler, IMAS based, June 2020

Dmitriy Yadykin on behalf of CPT



CHALMERS
UNIVERSITY OF TECHNOLOGY



This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 under grant agreement No 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.



- This tutorial is user(developer) oriented, no detailed mechanism description of the tools
- Useful features are shown and the differences between two environments (ITM/IMAS) are highlighted
- More items can be added following requests, these will be included in the final version (probably at the end of Code Camp)



- latest imasenv module: **imasenv/3.28.1/rc**
- *imasenv* is frequently updated (key modules)
- *imasenv* is 'compiler specific', versions for intel (*imasenv/3.28.1/intel/17.0/rc*) and gnu (*imasenv/3.28.1/gcc/7.3.0/rc*) compilers are separated (although not completely), default is intel version
- *kepler module* is the part of *imasenv module* (affects the way Kepler is used, more in Kepler section)
- tokamak 'agnostic' (no device should be specified by default)
- More on setting the environment:
 - <https://docs.psnc.pl/display/WFMS/IMAS%3A+Setting+working+environment>
 - https://users.euro-fusion.org/iterphysicswiki/index.php/Setup_for_ETS-6



- Latest version is **imas-fc2k/4.10.1**
- IMAS version of FC2K is now ahead of the ITM version (although usage is similar). IMAS fc2k is the main development branch now (new features are not included in the ITM version automatically)
- New features in the IMAS FC2K
 - python actor generation
 - validation of the code parameters
 - use of module/header during actor generation (recommended option)



Python actor generation:

Actor generation:

Generate Kepler actor Generate Python actor

- Python actor is generated in parallel to the Kepler one
- Two branches are separated
- Python actors are saved in the separate location (\$ITMWORK/PYTHON_ACTORS by default, can be modified in 'Environment' tab

Python actors:

More on python actors: <https://docs.psnc.pl/display/WFMS/Separation+of+Kepler+and+Python+actors>

Validation of the code parameters

Frequently Used XML:

Default XML:

XSD:

- Code parameters are validated automatically during actor generation
- Manual validation can be done also using 'Validate' button



Use module/header during actor generation

Source Code

Default library:
[Text Field] [Browse...]

Optional library:
[Text Field] [Browse...]

Use module(Fortran) / header(C++):
[Text Field] [Browse...]

‘Safer option that adds additional check on the routine consistency during actor generation (checks for argument consistency in the source code)

Other useful features in FC2K

- Sandbox (advanced option to store actor state during standalone execution)
- Diagnostic info (way to output warnings/errors during actor execution and enforce ‘soft’ stop of kepler execution)
- More information can be found here:
<https://docs.psnc.pl/display/WFMS/FC2K+actors+settings>



- Central 'naked' kepler is loaded as module in the imasenv (latest version is **kepler/2.5p5-3.1.1**)
- Dressed kepler(s) are also available as modules and can be listed (using **module avail kepler**) and loaded (using **module switch <dressed kepler>**); this is required each time imas environment is loaded
- Dressed kepler should be installed locally (using **kepler_install <local name>**) and loaded (using **kepler_load <local_name>**) to use it
- All local settings are kept inside local installation folder, and several keplers can be launched at the same time (this is also true for running IMAS/ITM versions in parallel). Be careful to use different run_out, run_work numbers
- Latest dressed kepler module is **kepler/2.5p5-3.1.1_dressed_3.28.1**
- More on the dressed kepler installation: https://users.eurofusion.org/iterphysicswiki/index.php/Setup_for_ETS-6#Installing_a_.22dress_Kepler.22



- AutoGui module is available to launch annotated workflows. This will be covered in separate tutorial
- Kepler has number of parameter that can be used when launching it; these can be used to run kepler in nogui mode, pass the particular workflow or parameter file, etc. Options can be listed using (**kepler -h**) in the terminal. Note: the meaning of some parameters is different between IMAS and ITM versions (for example *-force* option). Consult the description of the option or contact CPT if you are not sure
- Parameter files can be loaded inside the kepler canvas but can not be saved (autoGui has this option)



- Three channels:
 - ITER gira <https://jira.iter.org/>
 - GFORGE <https://gforge6.eufus.eu/gf/project/generalsupport/tracker>
 - ticketing system on cineca <http://ticket.eufus.eu/>
- ITER gira:
 - IMAS related
 - problems with Access Layer (Data Dictionary)
 - fc2k problems, kepler problems
 - IMASviz
- GFORGE:
 - ITM related
 - ETS related (both ETS 5 and 6)
 - IMASGO, IDS2CPO, CPO2IDS, visualization tools
 - account related problem (access rights, etc.)
- Cineca:
 - Gateway support
 - login problems
 - module problems
 - storage related problem (quota)