Kepler Wokflow management tool: autoGui

Dmitriy Yadykin on behalf of CPT

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.

Notes, content

- This tutorial is based on autoGui version 1.9 and ITM (autoGui is available for IMAS also)
- Contents:
  - general information
  - workflow configuration and execution
  - workflow annotation to be used in autoGui

General Description

- AutoGui tool is developed to make kepler workflow configuration more user friendly (as compared to Kepler canvas)
- AutoGui can be used with annotated kepler workflows
- Autogui configures workflow outside Kepler and launches Kepler after configuration is completed
- Autogui automatically saves configuration parameters in the parameter file that is passed to the Kepler together with the workflow
- Parameter set can be saved, loaded in the autoGui
- Workflow can be executed interactively (on login node) or send to the queue
- Execution monitoring tools are available

First step: launch autoGui

- load environment
- check that autoGui module is in the module list
  - if **ets module** is loaded then it is
  - if **itmenv module** is loaded it is not automatically true (starting from itmenv/ETS_4.10b.10_v5.8.0 autoGui is not the part of itmenv, do ***module load autoGui*** if needed)
- autoGui is launched by issuing ***autoGui*** command in the terminal, this will open new (‘naked’) autoGui session
- if **ets module** is loaded the ‘dressed’ version of autoGui can be loaded with ***ets*** command; this will launch autoGui and load ETS5 workflow with default set of parameters

Workflow configuration and execution

- The detailed guide on the workflow configuration and execution is available here:

https://users.euro-fusion.org/tfwiki/images/0/09/Ets_config_ag_v2.pdf

This link is also available through the ETS documentation web page:
https://users.euro-fusion.org/tfwiki/index.php/ETS_Documentation#Configuration_of_ETS_workflow_in_Kepler
and ETS user's guide web page:
https://portal.eufus.eu/twiki/bin/view/Main/User_Guide_accessing_JET_data

Workflow annotation (introduction)

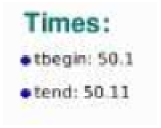
- Workflow need to be annotated to be ‘understood’ by autoGui
- Annotation is addition of the additional attribute to the parameters of the kepler workflow
- Any kepler workflow that is working under itm(imas)env can be annotated
- Workflow annotation is usually performed/controlled by the workflow manager

Workflow annotation (mechanism)

Parameter of the workflow is annotated by adding additional attribute to it

This can be done in two ways:

- editing workflow xml file
- adding attribute in the Kepler canvas (Note: additional workflow preparation is needed in general for this method to work)

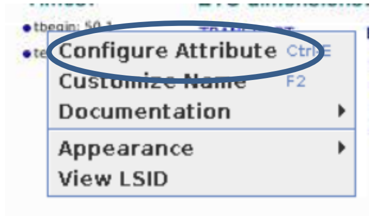

Times:
- tbegin: 50.1
- tend: 50.11

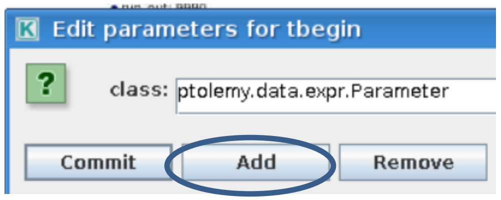
Example: annotation of the ***tbegin*** workflow parameter of ETS5 workflow

XML file:

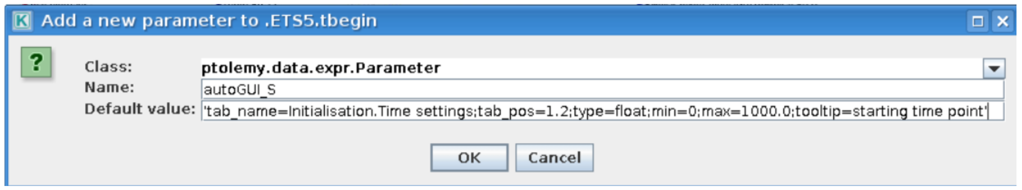
```
<property name="tbegin" class="ptolemy.data.expr.Parameter" value="50.1">
        <property name="entityId" class="org.kepler.moml.NamedObjId" value="urn:lsid:kepler-project.org/ns/:87028:71:1">
        </property>
….
        <property name="autoGUI_S" class="ptolemy.data.expr.Parameter" value="'tab_name=Initialisation.Time
settings;tab_pos=1.2;type=float;min=0;max=1000.0;tooltip=starting time point'">
        </property>
…
</property>
```

Workflow:





class: ptolemy.data.expr.Parameter



Class: ptolemy.data.expr.Parameter
Name: autoGUI_S
Default value: 'tab_name=Initialisation.Time settings;tab_pos=1.2;type=float;min=0;max=1000.0;tooltip=starting time point'

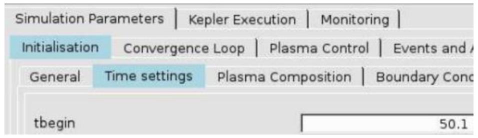
Value of autoGui attribute

Defines how autoGui will treat the parameter

'tab_name=Initialisation.Time settings;tab_pos=1.2;type=float;min=0;max=1000.0;tooltip=starting time point'

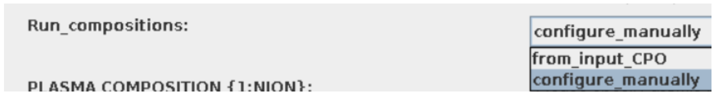
<u>tab name:</u> name of the tab, where parameter will appear

<u>tab_pos:</u> position of the tab in the autoGui



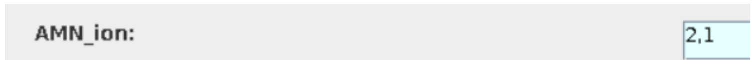
tbegin 50.1

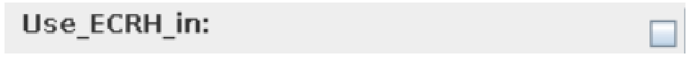
<u>type:</u> parameter type, possible types: int, float, string, vector, choice


Run_compositions: configure_manually

from_input_CPO
configure_manually

PLASMA COMPOSITION {1:NION}:


AMN_ion: 2,1

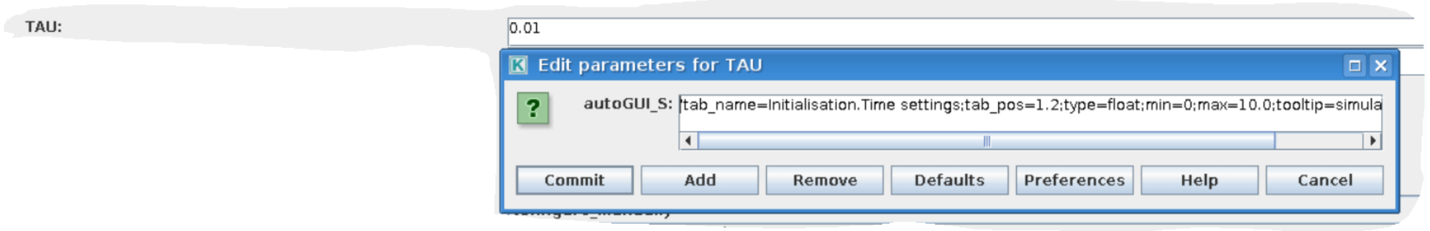

Use_ECRH_in: ☐

<u>min/max:</u> optional, limit the value of the parameter

<u>tooltip:</u> useful message to be displayed when pointing to the parameter in the autoGui

AutoGui Parameter types (property name)

autoGUI_S: simple parameter

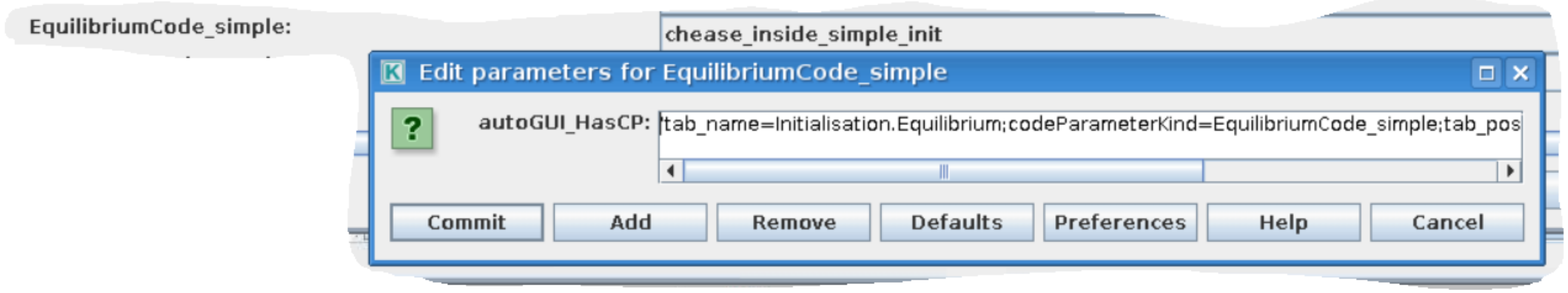


TAU: 0.01

Edit parameters for TAU

autoGUI_S: 'tab_name=Initialisation.Time settings;tab_pos=1.2;type=float;min=0;max=10.0;tooltip=simula

autoGUI_HasCP: parameter has actor with code specific parameters associated with it

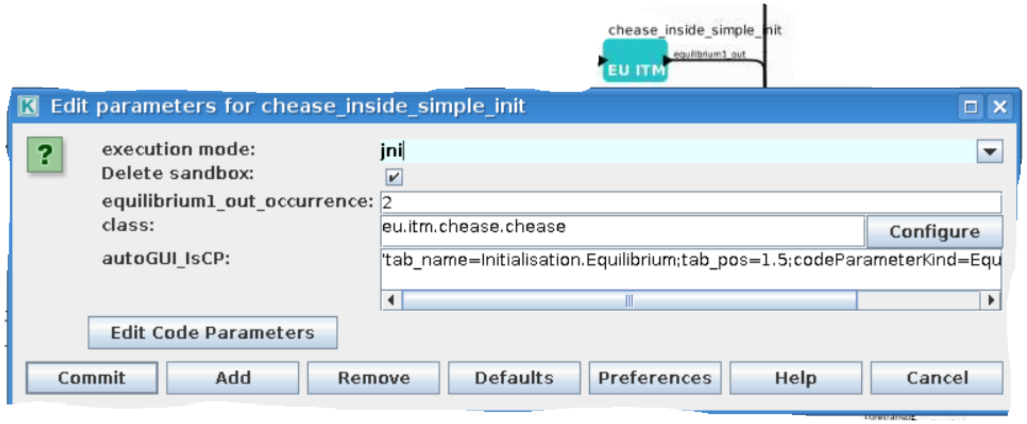


EquilibriumCode_simple: chease_inside_simple_init

Edit parameters for EquilibriumCode_simple

autoGUI_HasCP: 'tab_name=Initialisation.Equilibrium;codeParameterKind=EquilibriumCode_simple;tab_pos

autoGUI_IsCP: actor contains code specific parameters



Edit parameters for chease_inside_simple_init

execution mode: jni

Delete sandbox: ✓

equilibrium1_out_occurrence: 2

class: eu.itm.chease.chease

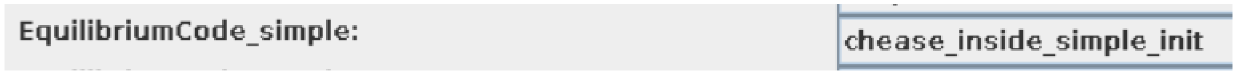
autoGUI_IsCP: 'tab_name=Initialisation.Equilibrium;tab_pos=1.5;codeParameterKind=Equ

Annotating actors with code specific parameters

Two parts should be annotated consistently: autoGUI_HasCP and autoGUI_IsCP

Simple example (name of the actor is the name in the choice)

autoGUI_HasCP



EquilibriumCode_simple: chease_inside_simple_init

'tab_name=Initialisation.Equilibrium;codeParameterKind=EquilibriumCode_simple;tab_pos=1.5;type=choice;tooltip=chease or other choice'

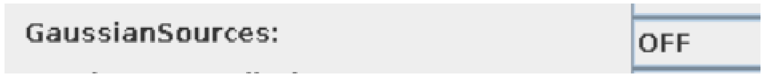
autoGUI_IsCP

'tab_name=Initialisation.Equilibrium;tab_pos=1.5;codeParameterKind=EquilibriumCode_simple'

codeParameterKind is used to build correspondence

More advanced example (name of the actor is not the name of the choice)

autoGUI_HasCP



GaussianSources: OFF

'tab_name=Convergence Loop.Sources.Main; tab_pos=2.4.1; codeParameterKind=GaussianSources; type=choice; tooltip= gaussian'

autoGUI_IsCP

'tab_name=Convergence Loop.Sources.Main; tab_pos=2.4.1; codeParameterKind=GaussianSources; correspondentChoice=ON'

correspondentChoice is used to define the value of the HasCP parameter when code parameters should be used