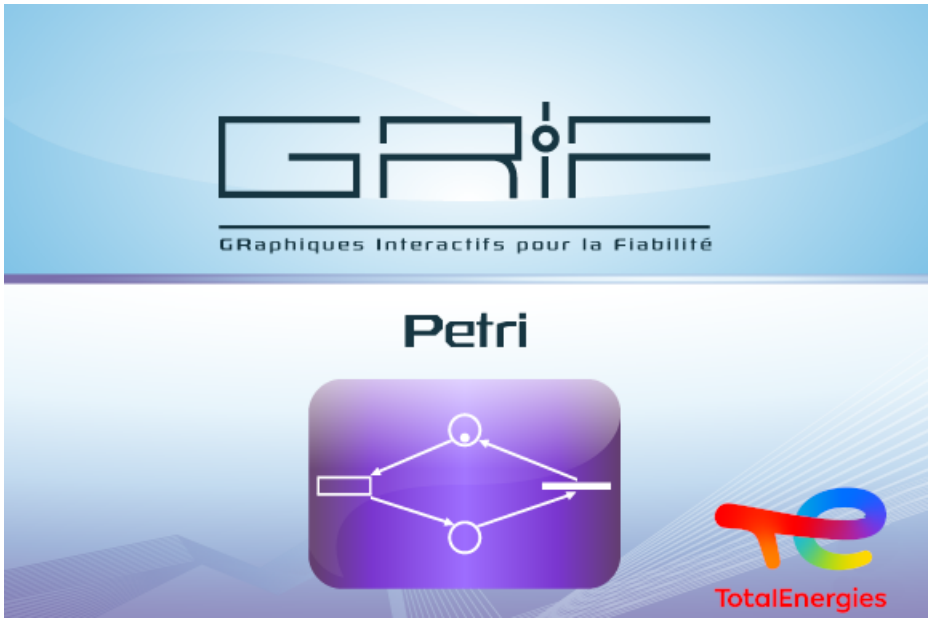
GRIF 2022

Petri Nets with predicates



User Manual

Version May 6th, 2022

Copyright © 2022 TotalEnergies

Abstract

This document is the user manual of *Petri Nets with predicates* module of *GRIF 2022*

Table of Contents

1. Description of the interface 6
1.1. Main window of the Petri Nets with predicates module 6
1.2. Vertical Tool bar 7
1.3. Data Editing Tables 7
1.4. Tree view 11

2. How to create a Petri Net 12
2.1. Entering the Net 12
2.1.1. Inputting Places 12
2.1.2. Inputting Transitions 12
2.1.3. Inputting Upstream and Downstream Arcs 13
2.1.4. Inputting Local Data 14
2.1.5. Entering Comments 15
2.2. Configuring the Elements 16
2.2.1. Configuring the Places 16
2.2.2. Configuring the Arcs 17
2.2.3. Configuring the Transitions 18
2.3. Petri Net example 22
2.4. Using repeated places (or shortcuts) 23
2.4.1. Search shorcuts 24
2.5. Page and group management 25

3. Menus presentation 27
3.1. File 27
3.2. Edit 28
3.3. Tools 29
3.4. Document 31
3.5. Data and Computations 31
3.5.1. Edit data tables 32
3.5.2. Parameters database 33
3.5.3. Batch computations 33
3.6. Simulation 33
3.7. Group 34
3.8. Utils 34
3.9. ? 35
3.9.1. Configuration 35

4. Data Entry Aids 37
4.1. Copy / Paste / Renumber (without shortcut) 37
4.2. Ordinary Copy/Paste 39
4.3. Overall change 39
4.4. Selection change 40
4.5. Alignment 40
4.6. Multiple selection 41
4.7. Selecting connex (adjacent) parts 41
4.8. Zoom and page size 41
4.9. Cross hair 42
4.10. Gluing/Associating graphics 42
4.11. Line 42
4.12. Table Cleaning 43
4.13. Prototypes 43
4.14. Document properties / Track change / Images management 43
4.15. Compare 2 documents 46
4.16. Files of the documents 47
4.17. Hypothesis 48
4.18. Picture Anchor 48

5. The parameters 50
5.1. Creation 50

6. Attributes 52
6.1. Creation 52
6.2. Use of the attributes 52

7. Table values 54
7.1. Creation 54
7.2. Use 55
7.3. Computations 56

8. Profile tables 57

9. Statistics and Setup of Variables 59
9.1. Definition of statistic states 59
9.2. Configuration of statistic states (or variables) 59
9.2.1. Types of statistics 60
9.2.2. Computation times 60
9.2.3. Histograms 61

10. Simulation interactive 62
10.1. Introduction 62
10.2. Simulation panel 62
10.3. Simulation history 63
10.4. Simulation scheduler 64
10.5. Configuring the simulation 64
10.6. Colour code / Legend 65
10.7. Transition firing 65
10.8. Automatic firing of transitions with zero delay 67
10.9. Probability firing for transitions with fire on demand 68
10.10. Simulation with groups 68
10.11. Dynamic fields 69

11. Computations 71
11.1. MOCA computations 71
11.1.1. Configuring the computations 71
11.1.2. Reading the results 75
11.2. Compute manager 76
11.3. Result Bank 77
11.4. Batch computation 78
11.5. Preparation of calculations 79

12. Curves 80
12.1. Edit curves window 80
12.2. Selection of results window 82
12.2.1. Curves from data in result-bank 82
12.2.2. Comparative curves from data in results bank 82
12.3. Examples of curves 83
12.3.1. Availability 83
12.3.2. Timing Chart 84
12.3.3. Fixed Size Histogram 84
12.3.4. Equiprobable Classes Histogram 85
12.3.5. Defined Interval Histogram 85

13. Database of parameters 87
13.1. Format of the databases 87
13.2. Connect to a database 87
13.2.1. Connection to a CSV file 88
13.2.2. Connection to a XLS file 89
13.2.3. Connection to a database (with a JDBC connection) 90
13.3. Import parameters from a connected database 91
13.4. Update of the parameters from the database 92
13.5. Rebuild of the links to the database 93

14. Save 94
14.1. Document template 94

14.2. RTF File 96
14.3. Results 96
14.4. Curves 98
14.5. Tables 98

15. Printing 99
15.1. Page setup 99
15.2. Print 99
15.3. Save in RTF file 104

16. Options of GRIF - Petri Nets with predicates 106
16.1. Options 106
16.2. Executables 106
16.3. Graphics 106
16.4. Digital format 107
16.5. Computations / Results 107
16.6. Places 107
16.7. Transitions 107
16.8. Arcs 108
16.9. Local data 108
16.10. Simulation 108
16.11. Curves 109

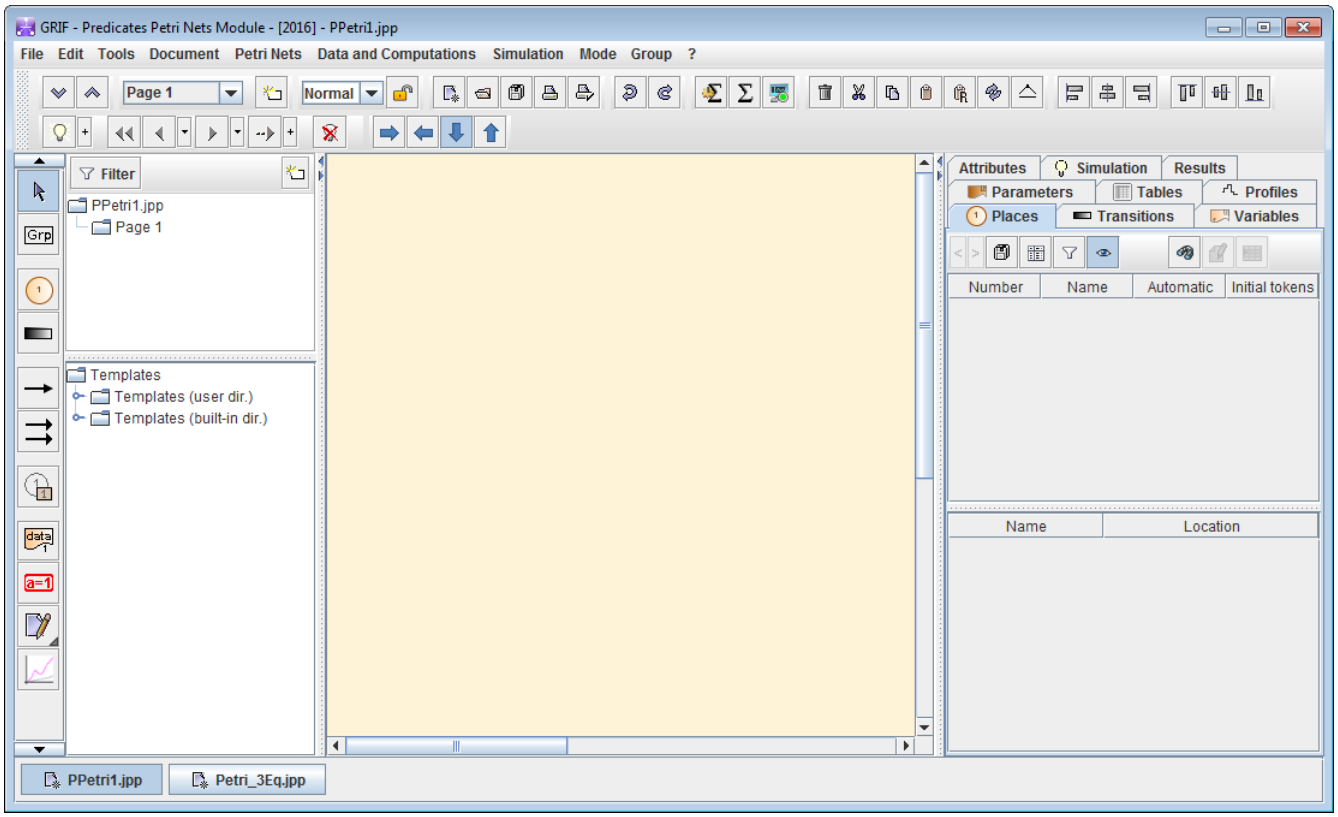
17. Annex 110
17.1. Hazard Rate h(t) Utils 110
17.1.1. Tables tab 111
17.1.2. Points tab 112
17.1.3. Polynomials tab 113
17.2. Moca seed testing utilitary 114

1. Description of the interface

1.1. Main window of the Petri Nets with predicates module

The main window is divided into several parts:

- **Title bar**: The title bar shows the names of the module and file being edited.
- **Menu bar**: The menu bar gives access to all the application's functions.
- **Icon bar (shortcuts)**: The shortcut bar is an icon bar (horizontal) which gives faster access to the most common functions.
- **Tool bar**: The tool bar (vertical) enables you to select the elements for modeling.
- **Input zone**: A maximum amount of space has been left for the graphical input zone for creating the model.
- **Tree**: A tree is between input zone and tool bar. It enables to walk through pages and groups of the document.
- **Template**: Templates are hidden behind the tree. They are grouped in two files following the save directory (User or built-in directory).
- **Set of tables**: Tables are gathered in "hidden" tabs on the right.



1.2. Vertical Tool bar

Each model used for operating safety has its own icons. All the Petri Nets graphical symbols are shown on the vertical icon bar on the left of the input window.

The vertical tool bar contains the following items:

- **Places** represented by circles.
- **Transitions** represented by rectangles.
- **Upstream and downstream arcs** represented by arrows.
- **Repeated place (Shortcut)** to create links between several parts of the same model (on different pages or in different groups).
- **Comment** to add text directly on the chart.
- **Affichage dynamique** to display the value of an element in the model.
- **Local data** create variables linked only to one part of the model.
- **Charts** to draw curves representing computations on the model.

- **Simulation** to switch to simulation mode.

1.3. Data Editing Tables

1. Description of the Tables

To create or modify data (parameters, variables, etc.), tables are available in the **Data and Computations menu** and in tabs at the right of the view. All the GRIF 2022 data tables operate in the same manner.

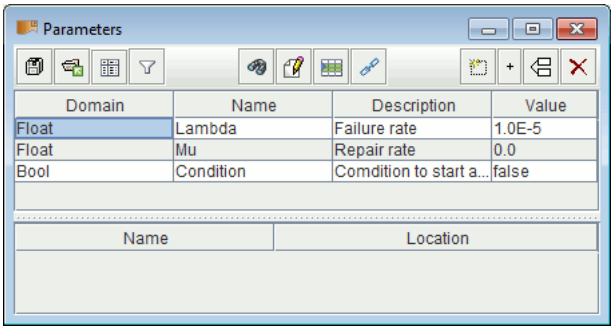
It is possible to edit all tables in another screen using **Data and Computations - Editing tables (new windows)** menu.

The data editing table/panel is divided into 3 parts:

- The upper part consists of a toolbar;

- The main part containing the data table.
- The bottom part indicating what the selected data is used for. This table is available only if the given data can be used by another data. The first column of this table indicates the name of these elements, the second indicates their location in the document (page, group). A click on a line from this lower table will open the page where the item is located and select it.

Here is an example illustrating the parameter table



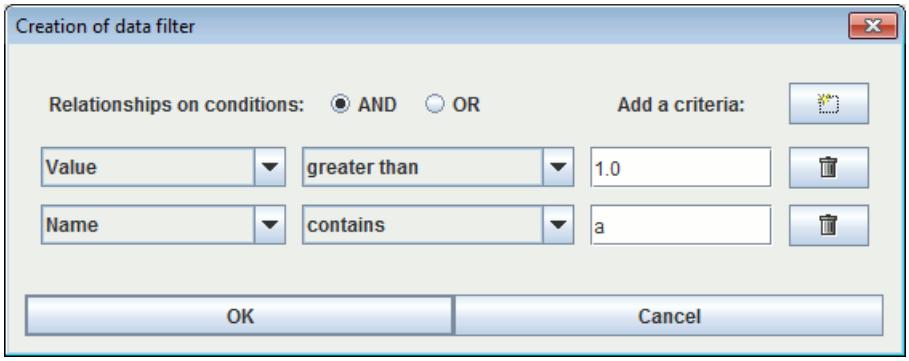
Different actions are available depending on the type of data displayed. Below is a non-exhaustive list of actions that can be found on the data tables.

	Saves the table in a text file.
	Import data from another Petri12 model or from CSV file.
	Opens the column manager (cf. Section 1.2, “Column manager”).
	Displays a panel for searching or filtering data (cf. Section 1.1, “Filter and sorting data”).
	When the **display selection** button is pressed, a click in the table leads to the selection in the input area.
	Find and/or replace expression in the table .
	Edit the selection.
	Multiple modifications made to all the selected data.
	Permit to merge data in a unique data.
	Creates new data.
+	Create the number of data indicated by user.
	Duplicate the selected data (ask a new name)
	Deletes the selected data (one or many).

1.1. Filter and sorting data

The filter panel enables you to display only what is necessary in the data table.

It consists of a search part: the text entered is searched in all the cells of the table, only the lines whose text is present are preserved; and an advanced filtering part allowing to consider finer criteria according to the different fields of the data. It is possible to combine several filtering criteria, as below:



Select **AND** or **OR** to choose the type of association between each line (filter criterion). A line is a Boolean expression divided into 3 parts:

1. the first is the column on which the filter is used;
2. the second is the comparator;
3. the third is the value to which the data will be compared.

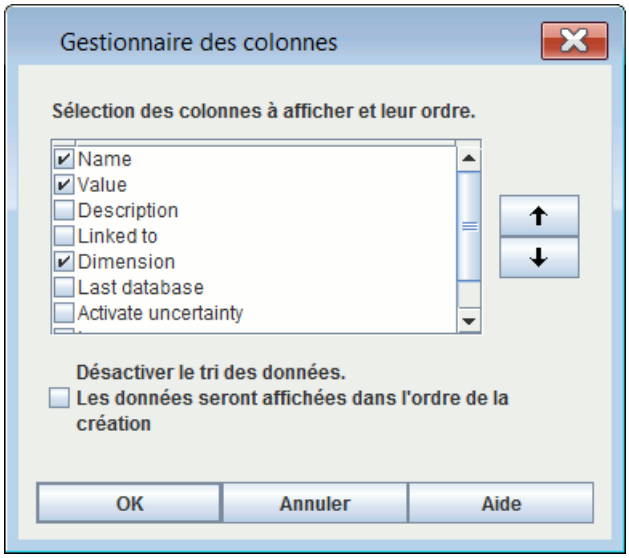
If the Boolean expression is true, the data will be kept (displayed); otherwise the data will be masked. When the filter is enabled its value is displayed between < and >.

The data in a column can be sorted by double clicking the header of this column. The first double click will sort the data in ascending order (small triangle pointing upwards). The second double click on the same header will sort the column in descending order (small triangle pointing downwards).

The choices that are made are kept on the current document. They will be reapplied when reopening your document and do not affect other documents in the application.

1.2. Column manager

A table can contain many columns and to improve its readability it is possible to choose the columns that will be displayed as well as their order. To do this, click on the **Columns Manager** button, the following window opens:

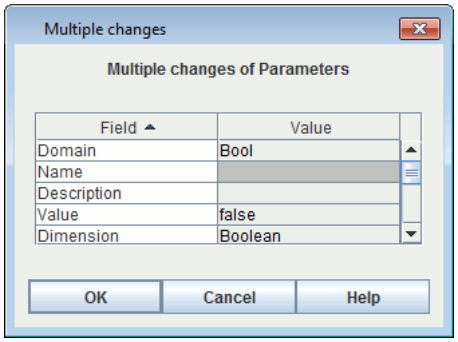


You can choose the columns to be displayed by selecting (or deselecting) the corresponding check boxes. The arrows on the right are used to move the columns up or down in the list to choose the order of the columns. The **Disable data sorting** check box disables the data sorting. This improves the application's performance with very complex models.

The choices that are made are kept on the current document. They will be reapplied when reopening your document and do not affect other documents in the application.

1.3. Multiple edition

To modify data, simply double-click on the cell to modify. When several lines are selected (using the CTRL or SHIFT keys) changes can be made to all the selected data by using **Multiple changes**. A window then opens to allow you to make these changes.

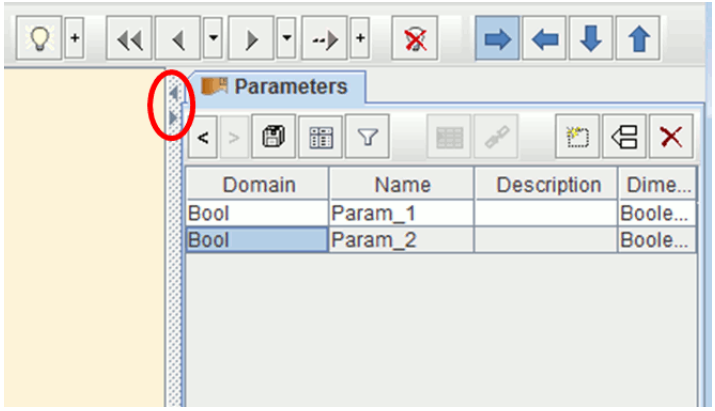


Items which cannot be modified are greyed. The white lines indicate that the selected data does not have the same value for the field in question. A new value can be entered which will be taken into account for all the selected data. The lines with no background colour indicate that all the selected data has the same value for this field (in this example the selected data is all "Float"); they can be changed to give a new value to all the selected data.

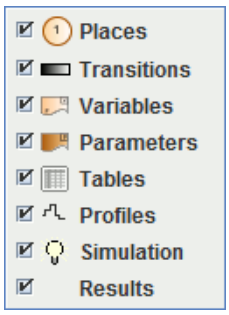
2. Table accessibility

As mentioned above, the tables can be accessed via the **Data and Computations** menu; in this case, each table is displayed in a separate window.

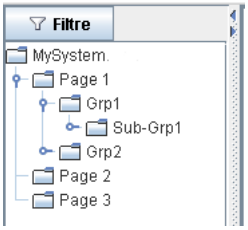
To avoid having too many windows open, all the tables are grouped together in tabs on the right-hand side of the application. This area can be hidden/displayed using the small arrows above the input zone.



It is possible to choose the tables in this zone by right clicking on the tabs. A contextual menu appears, in which the user can select the tables s/he wishes to display.



1.4. Tree view



To help users to walk through the document (pages, groups and sub-groups), a tree is available on the left of the application. By default, every element is displayed, you can use **Filter** button in order to select elements you want to display or not.

You can expand or collapse a node in a recursive way with a right click on the node.

As explained for tables on the right, you can "hide" the tree.

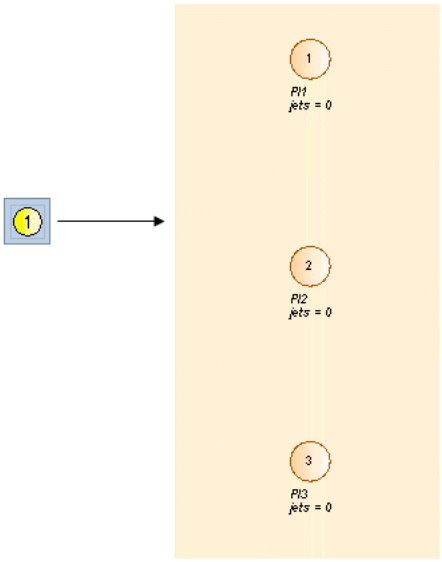
2. How to create a Petri Net

2.1. Entering the Net

2.1.1. Inputting Places

To input the different **Places**, select the corresponding symbol on the symbol bar. A new element is then created each time you click the mouse on the graphical input zone. Each of the places in the model has three parameters:

1. A **number**: Located in the centre of the places, they are automatically incremented. These numbers are the true identifiers of the places which will be used by the computation engine. This is why two places cannot have an identical number.
2. A **label**: A default label is assigned to each place ("Pli" for Place number "i"). Since each place normally has a very precise meaning for the user you are strongly recommended to assign a label which is more mnemonic than that given by default. This enables you to locate yourself better in the model and in the results file.
3. A number of **tokens**: By default it is equal to zero for each place created. In a Petri Net the presence (or not) of a token in a place normally corresponds to the presence (or not) of a specific status for one of the components in the system modelled by the Petri Net. All the tokens present at a given instant (Petri Net "marking") thus correspond to the global status of the system studied. The change in this "marking" is the dynamic aspect of the system.



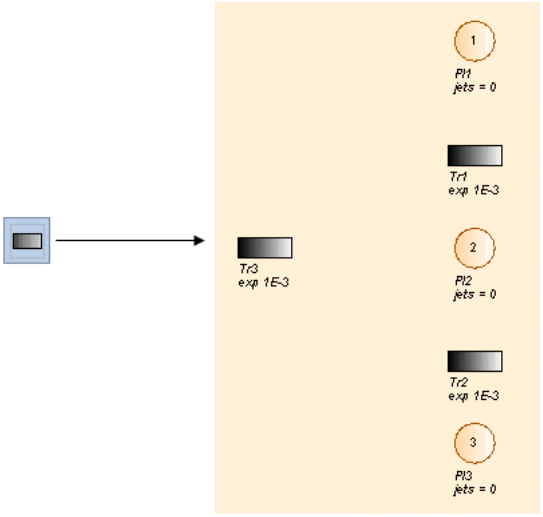
2.1.2. Inputting Transitions

To input the different **Transitions**, select the corresponding symbol on the symbol bar. A new element is then created each time you click the mouse in the graphical input zone.

In a Petri Net, the **Transitions** model the events which may happen at a given moment on the system studied (failures, tests, maintenance, etc.). Firing the transitions modifies the marking of the places to which they are linked by the arcs (upstream and downstream). This is used to simulate the system's dynamic behaviour.

When created, each transition has a default name ("Tri" for the transition entered in the ith position). Unlike the places, the transition number is unimportant since it is not used in the data file generated for the computation

engines. You are therefore highly recommended to assign them a mnemonic label (more than for the places) (cf. Section 2.2.3, “Configuring the Transitions”).



2.1.3. Inputting Upstream and Downstream Arcs

The "upstream arc" function describes part of the transition validation conditions (the other part is handled by the "guards" - cf. Section 2.2.3.4, “Guards tab”). They define the marking necessary for the upstream places to allow the transition to be fired.

The "downstream arc" function describes what happens at token transfer level when the transition is fired.

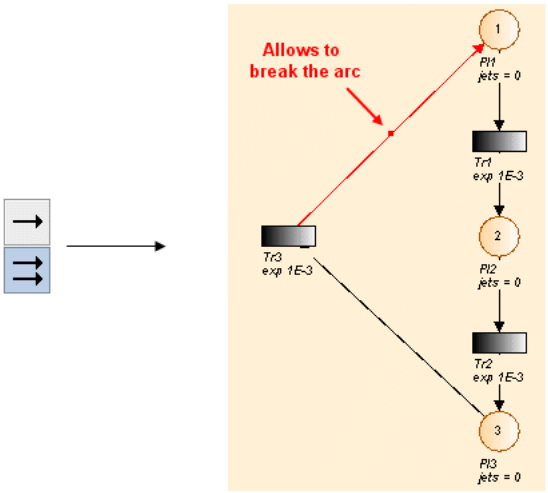
To input the upstream or downstream arcs:

1. select one of the two corresponding icons on the symbol bar:
   - the "single arrow" which is used to input only one arc at a time or
   - the "double arrow" which is used to input as many arcs as you wish.
2. select a start **Place** (respectively a **Transition**) by clicking on it with the left mouse button
3. drag the mouse (without releasing the button) to the final **Transition** (respectively the **Place**) and release the button.

It is the order "place" => "transition" or "transition" => "place" which determines the type ("upstream" or "downstream") of arc entered.

The result obtained is shown in the following figure. Upstream arcs have been drawn between place 1 and transition Tr1, then between place 2 and transition Tr2, and downstream arcs have been drawn between transition TR1 and place 2, then between transition Tr2 and place 3, etc. It must be noted that, unlike the "Reliability Networks, there are no bidirectional arcs for the "Petri Nets". However, it is often the case that a downstream arc must be drawn

between the same place and the same transition. In this case they can be superimposed and give the illusion that they are a bidirectional arc but in fact they are two separate arcs.

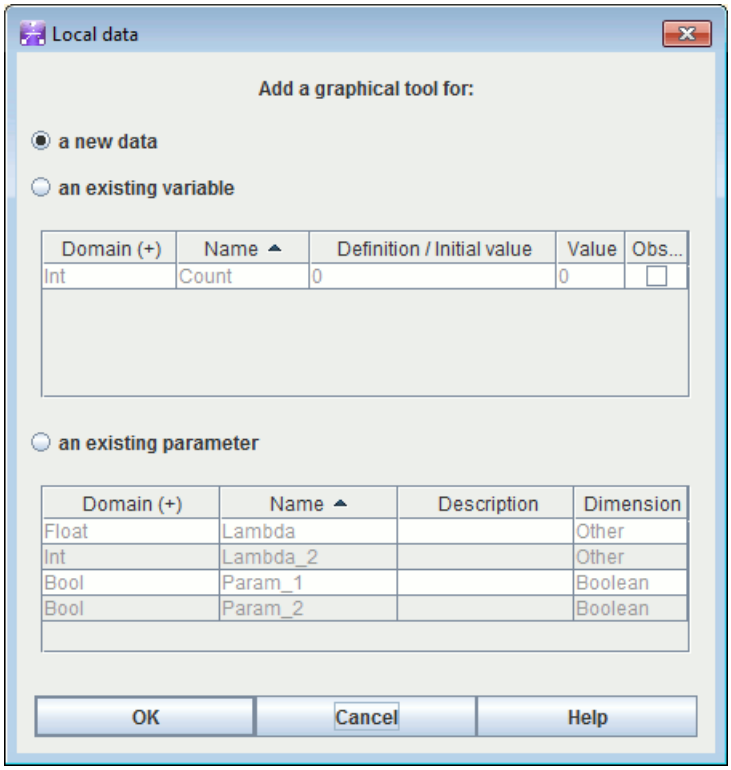


Notes:

1. For reasons of model appearance or legibility it may be useful to break down an arc into several parts. To do this, select the arc then move the small red square located in the middle of the segment.
2. It is also possible to straighten an arc using the command: **Tools - Straighten arcs** .

2.1.4. Inputting Local Data

To add local data to the model, select the corresponding icon on the task bar then click left at the point on the model where the data is to be placed. A window then opens:

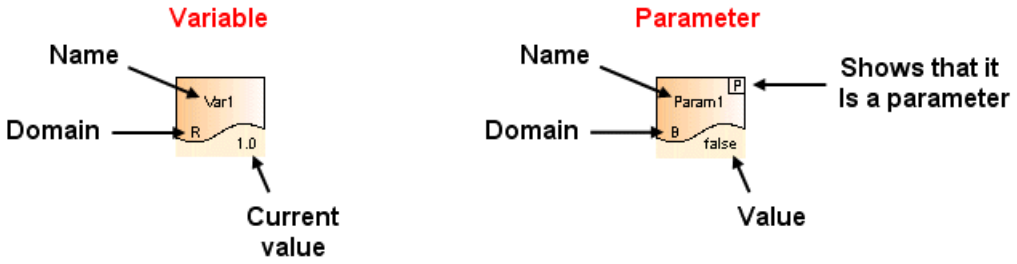


This dialogue box gives three choices:

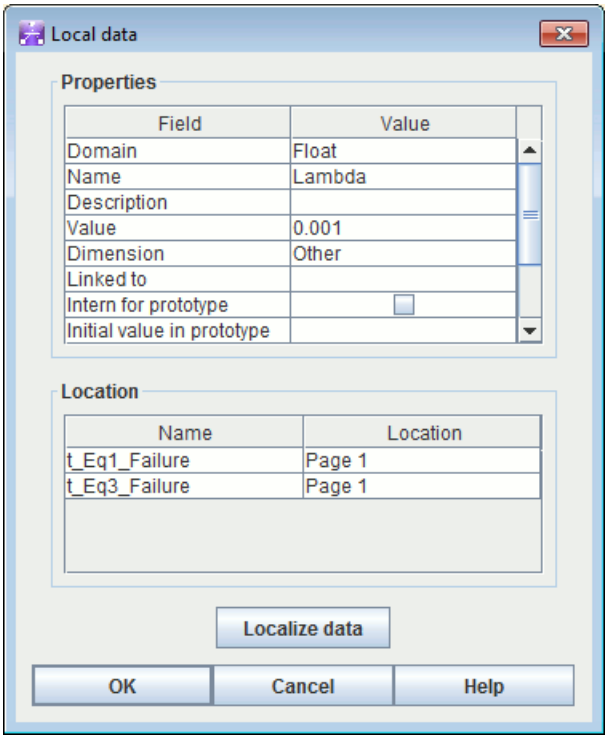
1. Create **new data**: opens a window where then you can create new data or a new parameter.
2. Use **an existing variable**: creates a graphical representation of a variable.

3. Use **an existing parameter**: creates a graphical representation of a parameter.

Once created, local data is represented as follows:



Click right on the local data to access its properties. Some fields can be modified in this manner: Name, domain, initial value, etc.

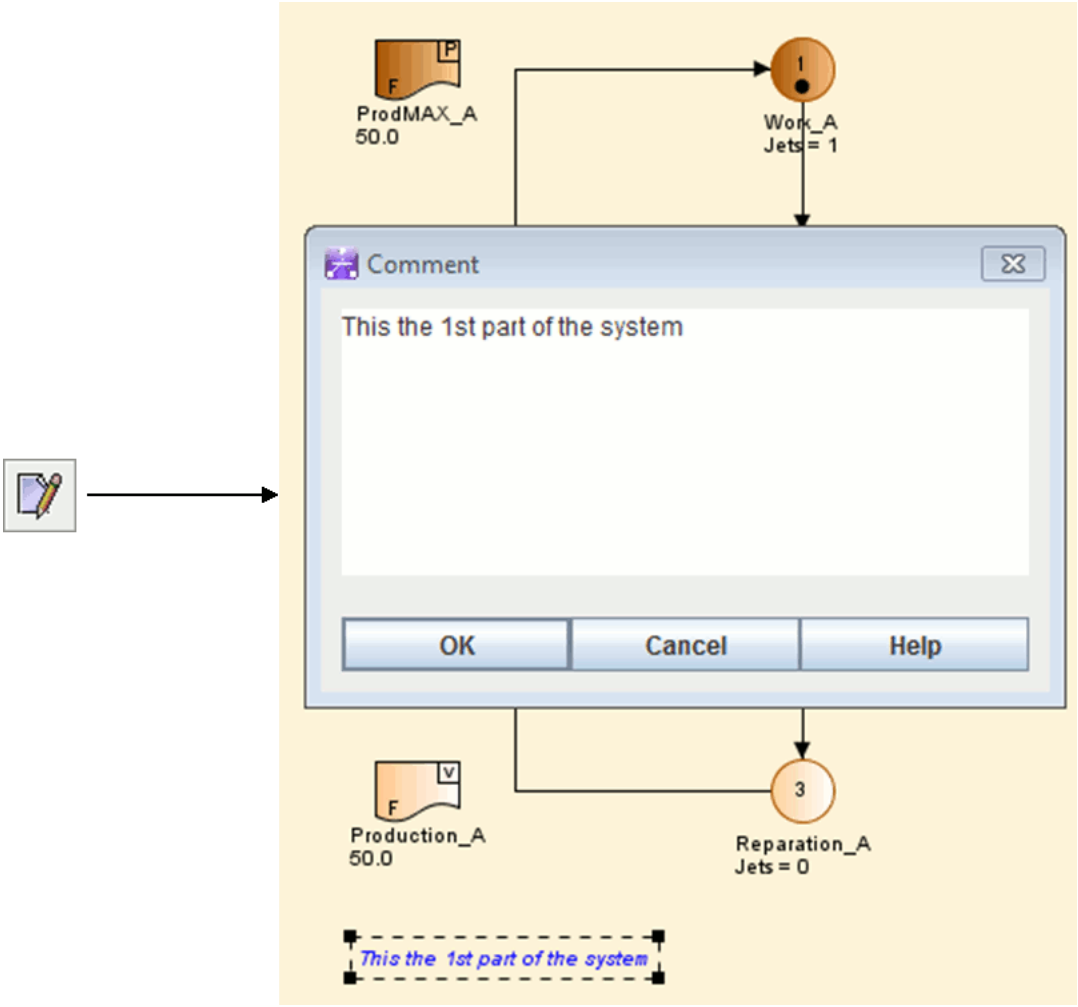


The fields where the term "prototype" appears are only used to create a prototype (cf. appended document on prototypes).

2.1.5. Entering Comments

To add a comment anywhere on the model, simply click on the pencil icon and position the cursor in a graphical entry zone. The **Comment** dialog box opens and you can then enter your comment.

The "%" sign is a special character and you have to type it twice for a single "%" to be displayed.



2.2. Configuring the Elements

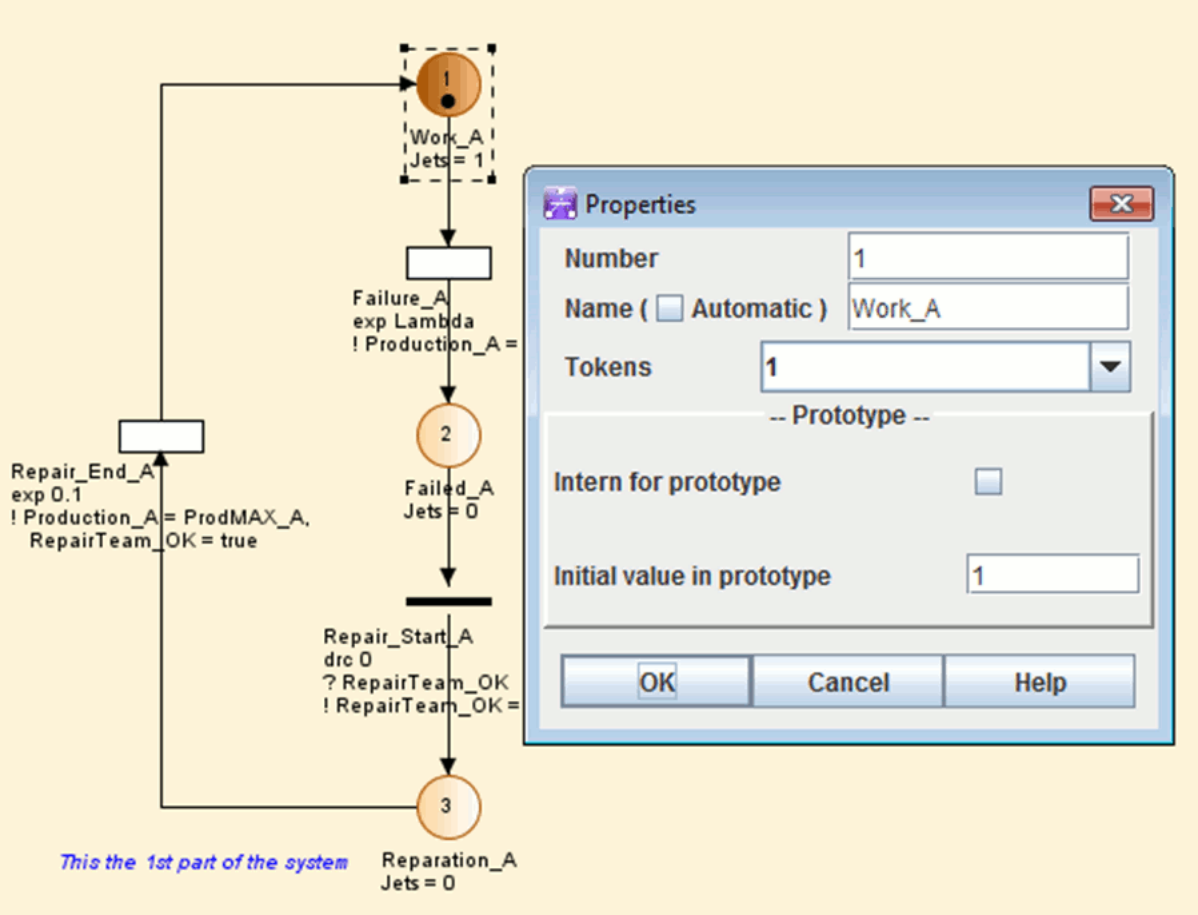
All the graphical elements can normally be edited with a double-click on them or using the **Edit - Properties** menu, or using the shortcut **Alt + Enter** .

2.2.1. Configuring the Places

The various place parameters are entered as follows:

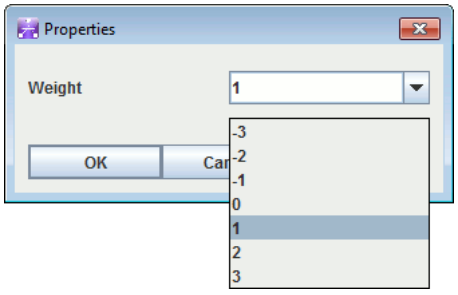
- Select the place using the right mouse button. A small panel then opens containing three fields to be filled in.
- Enter the place **Name** (optional but highly recommended). This description is "Pli" by default.
- Where necessary, change (carefully) the place **Number** . Other safer methods are described below.
- Enter the number of **Tokens** initially present in the place in question. This can be done manually but clicking the small black triangle is much easier.

The **Prototype** part is only used to create prototypes (see prototypes appendix).



2.2.2. Configuring the Arcs

By default, the **Weight** of all the arcs (upstream and downstream) is "1". However, this can be modified. To do this, click right on the arc relevant concerned to display the editor shown below.



Clicking the small black triangle displays some possible values which can be selected with the mouse. For the downstream arcs, the weights are always positive and correspond to the number of tokens which will be added to the downstream place when the corresponding transition is fired. For the upstream arcs there are three cases:

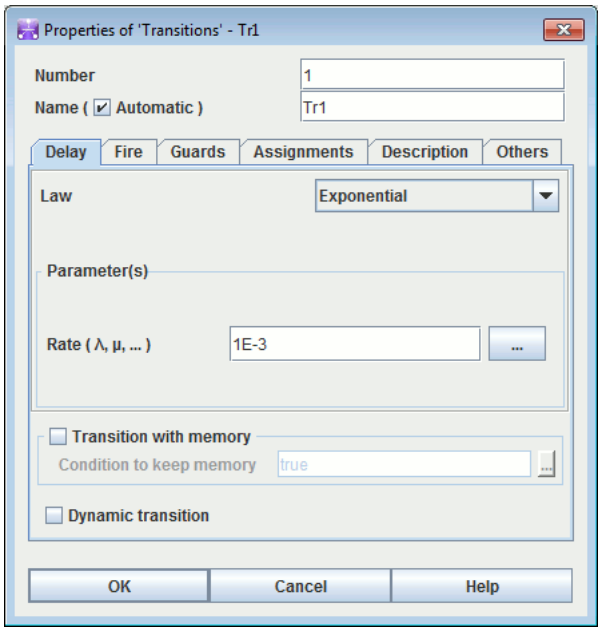
- Weight strictly positive: these are "normal" arcs which validate the transition when the number of tokens in the upstream place is greater than or equal to the weight of this arc. When the transition is fired, a number of tokens equal to the weight of the arc will be removed from the corresponding upstream place.
- Weight strictly negative: these are "inhibitor" arcs which inhibit the transition when the number of tokens in the corresponding upstream place exceeds the absolute value of the arc weight (e.g. 3 tokens for a weight of -3). This type of arc is graphically represented by a dotted line and does not modify the marking of the upstream place when the transition is fired.
- Weight "0": it is an arc which empties the corresponding upstream place when the transition is fired (whatever the marking of the upstream place before it is fired).

2.2.3. Configuring the Transitions

Entering the various transition parameters is more complex than those of places since it requires a good understanding of the MOCA-RP application's possibilities. You are highly recommended to consult the manual at this stage of the operations!

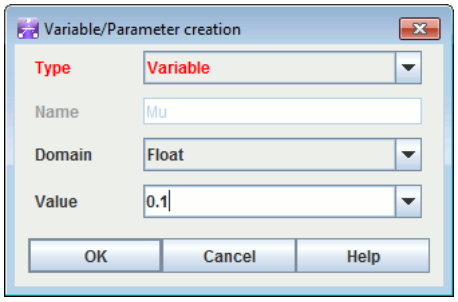
Click right to select the transition concerned. A dialogue window appears. The top part enables you to modify the **Name** and **ID** of the transition. The other part consists of 5 tabs for configuring the transition's behaviour.

2.2.3.1. Delay tab



The **Delay** tab indicates the transition delay law. Select the desired law, and then enter the parameters of the law. In this example the default law is "Exponential" with 1E-3 as a parameter. For the parameters, you can enter a value, a name (a parameter) existing in the document, or a new name. In this last case the following window will appear to ask for the value and the domain. In the example, Lambda is a **Real Parameter** (invariable) with value 0.001.

It is also possible to use a **Variable** in the case of dynamic reliability.



The **Transition with memory** option: Once the transition has been configured you still have to specify what happens when the transition is "inhibited" before it can effectively be "fired". This parameter is very important and two very different cases must be considered:

- Case 1: When the transition is inhibited, this indicates that the corresponding event can never happen (e.g.: an "old" component is replaced by a "new" one before it fails ==> the old component can now never fail). In this case, when the transition becomes valid again, it represents a new event (e.g.: failure of the "new" component) which has nothing to do with that which was "nipped in the bud" before it could happen. It is thus necessary to simulate a new firing delay for the transition: the transition has "no memory" of what happened previously.
- Case 2: When the transition is inhibited, this only indicates that the corresponding event is temporarily "suspended" (e.g.: a component repair is stopped due to it being night time ==> the repair will continue where it left off the next morning). In this case, when the transition becomes valid again, this only means that the delay relative to the suspended event continues. We must therefore use the delay which remained at the time when the

transition was inhibited as the new delay before the transition is fired: the transition must retain the "memory" of this residual delay. The choice is made by selecting (or not selecting) the Transition with memory check box.

You may need to erase memory, do to this you can input a **condition to keep memory** . If this condition (an expression) get "false", the memory is "lost" and a new delay will be computed the next time the transition becomes valid.

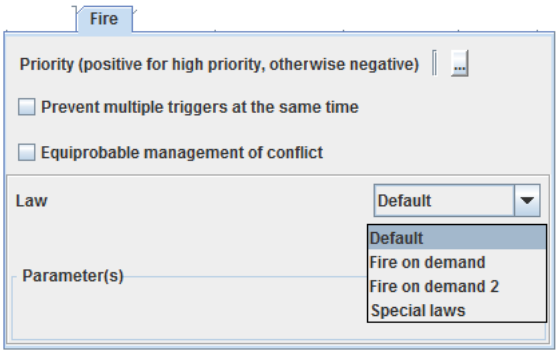
Dynamic transition option enables to recalculate the firing delay of the transitions even if the transition is not validated. Indeed, without this option when the lambda of the law changes, the delay is not revaluated as long as the transition is not invalidated and then re -validated. A dynamic transition has the peculiarity to recalculate are its delay when the parameters of its law are modified. The new delay is calculated in function of the duration already spent.

2.2.3.2. Laws of Delays

There is multiples laws available to represent a delay on a transistor, listed below (the laws from MOCA-RP, so directly explained in the documentation of MOCA-RP, are annotated with an asterisk *):

- The exponential law*
- The exponential law + WOW*
- The law "In advance schedule times"*
- The law "In advance appointed times"*
- The normal log law*
- The special laws*
- The uniform law*
- The triangular law*
- The Weibull1 law*
- The Weilbull2 law*
- The truncated Weibull1 law*
- The truncated Weibull2 law, which takes the truncated Weibull1 law, but where we inform the scale parameter eta, instead of the Mean MTTF parameter, as the Weibull2 law.
- The Acc exponential law, used by TotalEnergies (do not use it).
- The Cumulative distribution function law, which enables to create a law from a list of pairs, associating a time to an occurrence probability.
- The Distribution function (Hazard rate) law, which enables to create a law from a list of pairs, associating a time to a failure rate.
- The Advanced distribution function (Hazard Rate) law, which enables to create a law which, from a list of pairs, associating a time to a failure rate, as well as a factor q and a polynomial order, will create a classic distribution function law with an additional factor q. A visualization tool is available when clicking on the button "Advanced...", detailled in the Annex part.

2.2.3.3. Fire tab



The **Priority** check box is used to give the transition a priority level. If two transitions can be fired at a given instant, the one with the higher priority will be fired first. If the priority level is identical the transitions will be fired in the chronological order of their creation. Since 2010 version, priority can be a complex expression.

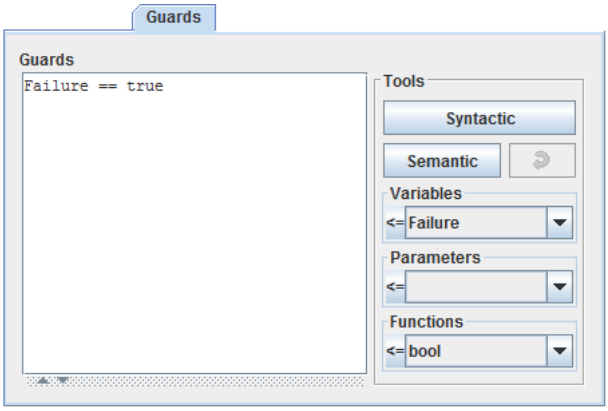
The **Prevent multiple triggers at the same time** prevents from multiple Drc 0 or IFA/IPA firing without time increase.

The **Equiprobable management of conflict** provides an equiproble firing frequency of 2 or more transitions when they are in conflict.

The **Fire** tab enables you to select the firing law for the transition:

- The **Default** law corresponds to the "normal" operation of the Petri Nets: the downstream places will be filled as defined in Section 2.2.2, “Configuring the Arcs” .
- The **On demand fire** law corresponds to the Moca-RP law of the same name: only one of the downstream places is filled after firing. The arguments of this law are the (N-1)th probabilities of happening in one of the N downstream places. The last probability is computed by the computation engine by making the 1s complement (see "Moca-RP User's Manual" for more details).
- The **On demand fire** law corresponds to the Moca-RP law of the same name: only one of the downstream places is filled after firing. The arguments of this law are the N-1 couple (probability,place). It is possible to specify probability to go in a given place. There are N-1 couple because last couple is computed by making the 1s complement, and using the place which is not selected. (see "Moca-RP User's Manual" for more details).
- The **Special laws** can only be used in the very special case where the computation engine has been recompiled to take account of it (Cf. Moca-RP User's Manual).

2.2.3.4. Guards tab



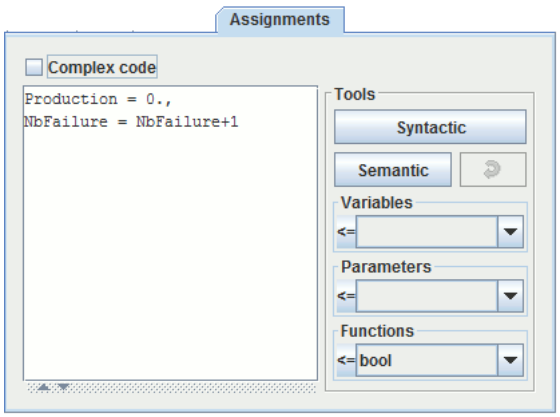
The **Guard** tab consists of a code editor where you can enter one (or more) guard(s) for the transition. A guard is a Boolean expression. The transition can only be fired if the guard is true.

The code editor has three parts. The first is an editable text zone where you enter the code using Moca-RP syntax. Under this zone is a noneditable zone indicating any errors which may arise. The third is the **Tools** part which is a data entry aid.

The Syntax button makes a syntax change. The Semantics button checks the semantics. The errors are displayed in the bottom left part. Under the buttons there are drop-down menus giving access to the model's various data. Select the desired data then click the <= button to insert it in the code.

The **Functions** drop-down menu shows funtions that can be used in Moca (cf. Moca User Manual).

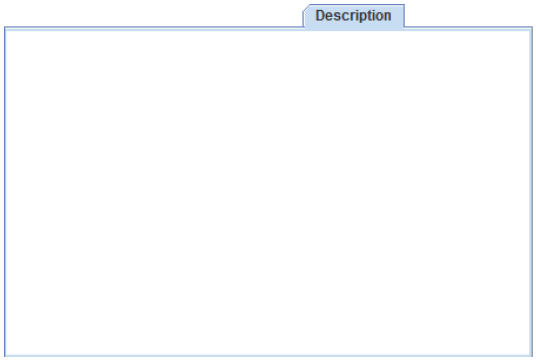
2.2.3.5. Assignments tab



The **Assignments** tab contains a code editor (identical to that of the Guards tab) for entering the assignments. The assignments are "played" when the transition is fired.

In Moca engine, usual behavior is the following: assignments are separate with "," and they are processed in parallel. You can ask for sequential processing of assignments using **Sequential assignments** check-box. In this case, each assignment must end with a ";".

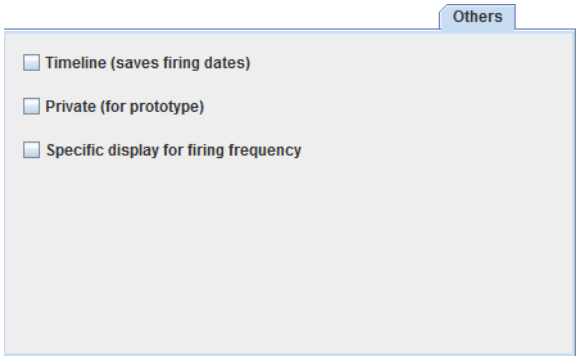
2.2.3.6. Description tab



The **Assignments** tab contains a code editor (identical to that of the Guards tab) for entering the assignments. The assignments are "played" when the transition is fired.

In Moca engine, usual behavior is the following: assignments are separate with "," and they are processed in parallel. You can ask for sequential processing of assignments using **Sequential assignments** check-box. In this case, each assignment must end with a ";".

2.2.3.7. Others tab



The **Others** tab contains 3 options:

1. Selecting the **Timeline** check box tells MOCA-RP to record all this transition's firing instants and to print them later.
2. The **Private** check box is only used to create prototypes (cf. prototypes appendix).
3. The **Specific display for firing frequency** enables to make show in a independent tab the firing frequency of the transitions for which the box is checked.

2.2.3.8. Addition of guards

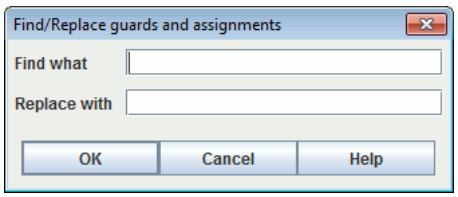
Once the transition's guards configured, it is possible to add one or more other guards. This functionality is available in the transitions table (**Edit transitions** tab, located in the right side of the application). To add one or more guards to one or more transitions, all we have to do is to select transition(s) to modify, then do a rigth click and finally select the **Add guard - With a "And"** menu or the **Add guard - With a "Or"** menu. **Add guard - With a "And"** menu will add guard(s) to the selected transitions by doing a "logical And" with the existing guards. **Add guard - With a "or"** menu will add guard(s) to the selected transitions by doing a "logical Or" with the existing guards. Gaurds to add are entered thanks to a code editor.

2.2.3.9. Addition of assignments

Once the transition's assignments configured, it is possible to add one or more other assignments. This functionality is available in the transitions table (**Edit transitions** tab, located in the right side of the application). To add one or more assignments to one or more transitions, all we have to do is to select transition(s) to modify, then do a rigth click and finally select the **Add assignment** menu. Assignments to add are entered thanks to a code editor.

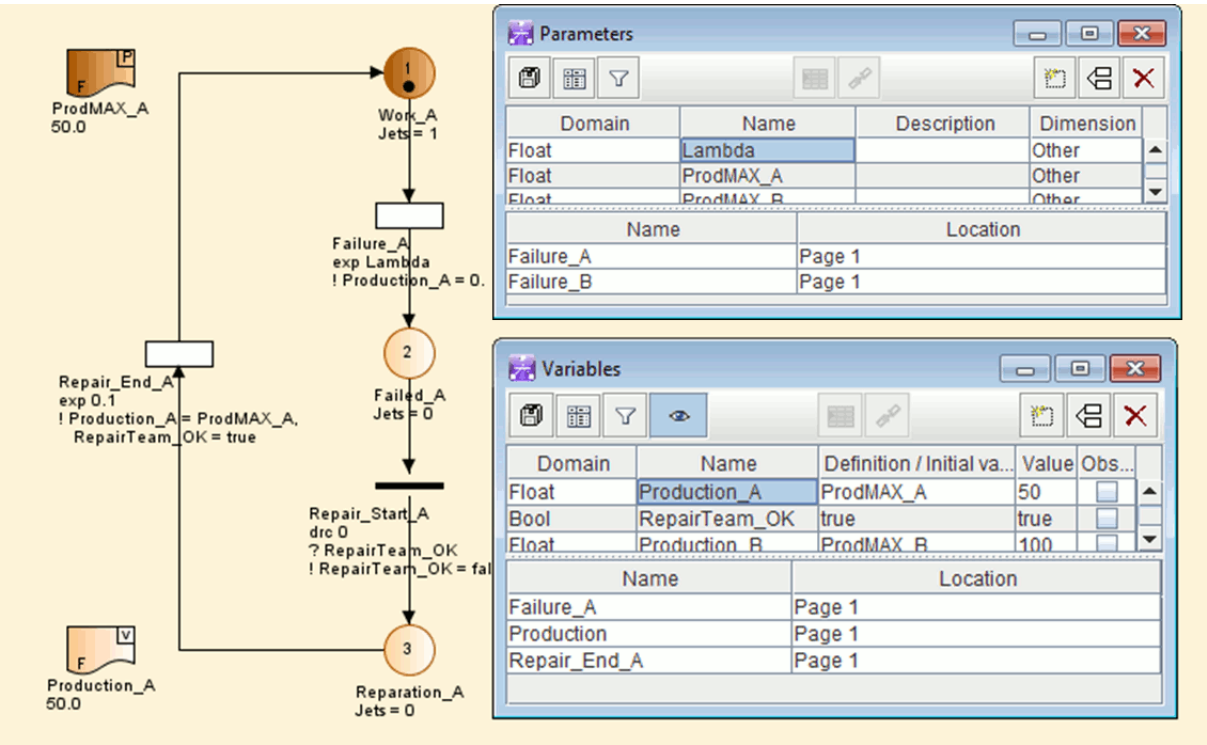
2.2.3.10. Find / replace on guards and assignments

Once the transitions configured, it is possible to perform a **Find / Replace** operation on guards and assignments. This functionality is available in the transitions table (**Edit transitions** tab, located in the right side of the application). To find and replace a string in the guards and assignments of one or more transitions, all we have to do is to select transition(s) to modify, then do a rigth click and select the **Find / Replace** menu and finally fill the following window:



- **Find what:** string to replace.
- **Replace with:** replacement string.

2.3. Petri Net example



The small Petri Net above represents the behaviour of a piece of equipment repaired by a maintenance team which is not necessarily available when the equipment fails.

This net has three places:

- **Work**: operating (place 1)

- **Failed**: failed, awaiting repair (place 2)
- **Repair**: being repaired (place3)

And three transitions:

- **Failure**: failure of the equipment
- **Repair_Start**: the equipment will be repaired
- **Repair_End**: the equipment is repaired and restarts

Here is how the model can be used to simulate the behaviour of a real piece of equipment

1. The **Work** place initially contains a token and the result is that the Failure transition is the only transition *valid* at the initial instant.
2. It will be fired when the component fails (delay fired randomly according to the exponential law assigned to this transition). The effect of this will be to remove the token from the **Work** place and to place one in the Failed place. In addition the **Production** variable will be reset to 0.
3. Since the arrival of the token in the **Failed** place is not sufficient to validate the **Repair_Start** transition, we must wait until the **RepairTeam_OK** variable (the message) input (or guard) to this transition becomes TRUE.
4. When the team of repairers is available (**RepairTeam_OK** message changes to TRUE) then the repair will immediately start since the delay law for this transition is a Dirac law with a zero delay.
5. When the **Repair_Start** transition is fired, the token is removed from the **Failed** place, a token is placed in the **Repair** place and the **RepairTeam_OK** message changes to FALSE (meaning "Repairers Unavailable"). Therefore, if another piece of equipment fails it must wait until the team of repairers is free before it can be repaired.
6. The arrival of the token in the **Repair** place validates the **Repair_End** transition which will be fired at the end of the repair delay (delay fired randomly according to the exponential law assigned to this transition).
7. When the **Repair_End** transition is fired, this removes the token from the **Repair** place, places a token in the **Work** place and again changes the **RepairTeam_OK** message to TRUE. The **Production** variable takes the **ProdMAX** value (100). Therefore we have returned to the initial state and the equipment is ready for the simulation of its second failure, etc.

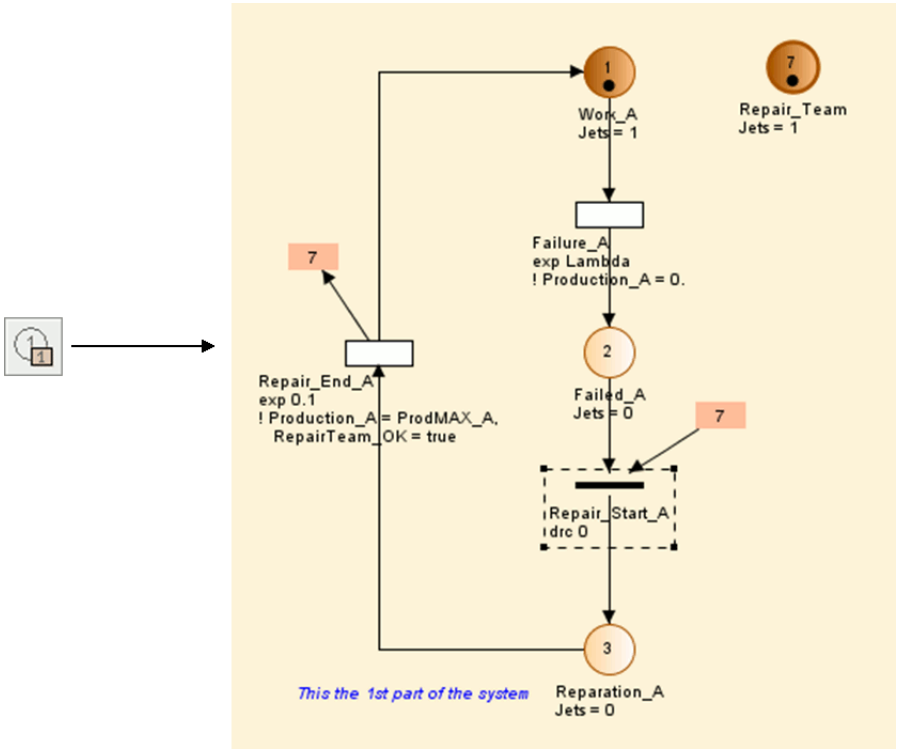
We have taken exponential laws but any other type of laws could have been used (e.g.: the log-normal law for the duration of the repair). Also, the Dirac law has enabled us to mix a deterministic phenomenon with random phenomena without hesitation. Therefore, although it is very simple, this small model already gives an idea of the power of the Petri Nets associated with the Monte Carlo simulation.

2.4. Using repeated places (or shortcuts)

The concept of a **shortcut** (or repeated element) was introduced in the Petri Nets with predicates module for four main reasons:

- To link together portions of the model;
- To avoid graphically complex model, and keep readability;
- To simplify the use of the **Group** function (cf. below);

- To highlight what is essential and what is not.



In order to create a **repeated place**, select the **repeated place** icon in the toolbar, then click on place (here place number 4). The **repeated place** (or shortcut) is displayed as a number if place has no token, or as a colored rectangle if place contains tokens.

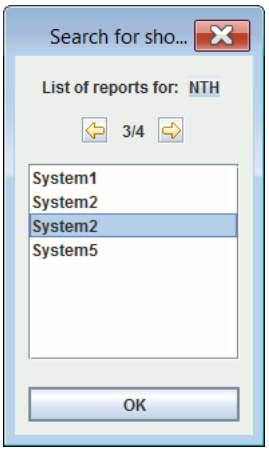
Then, the repeated place can be selected to be put where tou want. You link the shorcut to a transition with an arc. Operation can be repeated many times (2 times on figure).

The behavior of this Petri Net is the same as the one describe in above paragraph. Messages have been replaced by **Repair_Team** place. For the net describing equipment behavior, the **Repair_Team** place is auxiliary.

2.4.1. Search shorcuts

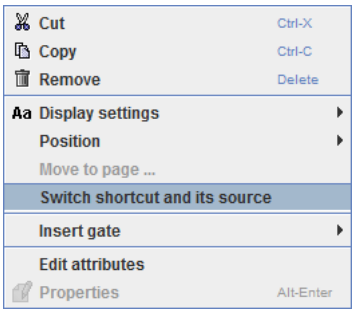
User can navigate between an element's different shortcuts, using **Search shorcuts**. This function is available in the contextual menu (with a right click on the element) or in the application menu **Search shorcuts**.

A window opens and displays the list of shortcuts.



Clicking on a shortcut automatically positions the view on this shortcut. You can return to the original element by clicking on its name at the top of the window.

In the shortcut contextual menu obtained with a right click, it is possible to swich the shortcut and its source.



2.5. Page and group management

The use of shortcuts allowed us to obtain two Petri Nets which have no graphical link between them. They communicate only by **shortcuts**. This can be used, for example, to place each subpart on a different page:

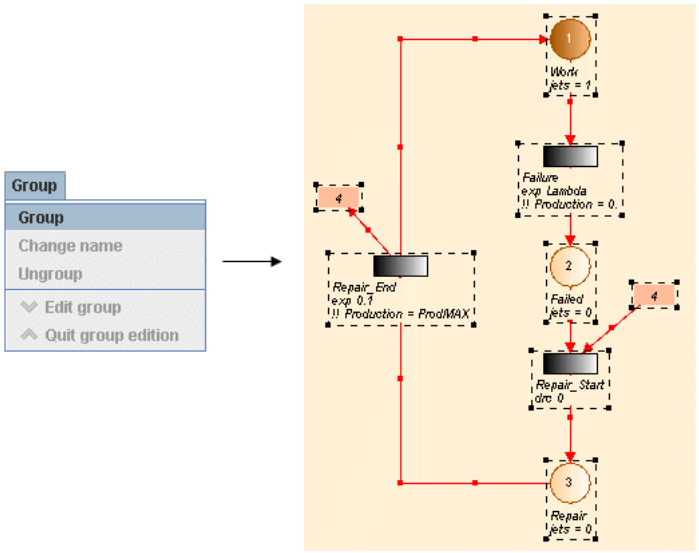
1. Create a new page by clicking the corresponding icon in the icon bar (or use menu **Tools - New Page**). A page number 2 is thus created.
2. Return to page 1 by selecting the page using the page selector in the ideographic command bar (or use menu **Tools - Page manager**).
3. Select the part to be moved.
4. Open menu **Tools - Change page**.
5. Select page 2 and click **OK**. The part selected is transferred to page 2 but it continues to communicate with page 1 via the **shortcuts**.

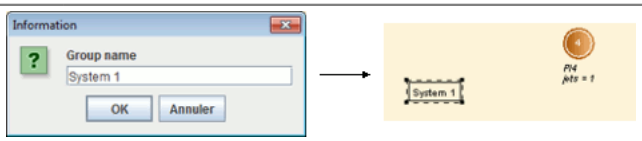
For large models the division method described above is very useful.

Another possibility for entering large Petri Nets is to use the **Group** concept. This is made possible by the **shortcuts** and the fact that the data is global for a document. This enables quite separate subparts to be created:

1. Select a subpart.
2. Use menu **Group - Group**. A dialogue box then opens asking for the name to be given to the group being created.
3. Enter the desired name and click **OK** (e.g.: "System 1"). The group is created: the subnet is replaced by a rectangle assigned with the chosen name.

You can also create an empty group with **Group - New Group** menu or group tool in the left toolbar.



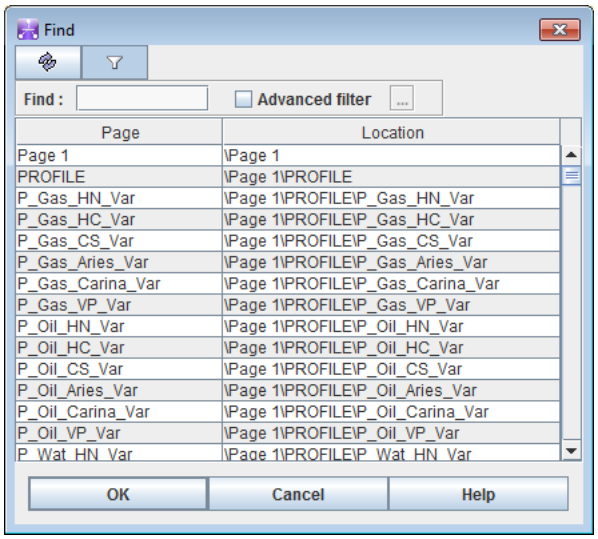


With a right click on the group, it is possible to view inside the group if **Overview of the contents** is checked.

Each group can then be **edited**, **renamed** or **ungrouped** using the commands in the **Group** menu. The group can also be edited with a click right or using the "cursor down arrow" on the left of the page manager. In Edit mode, the submodel can then be modified as you wish. When the modification is terminated you return to the previous figure by exiting group editing by menu **Group - Quit Group Edition**, or using the "cursor up arrow" on the left of the page manager. It's also possible to choose a picture for a group by using **Group - Change Picture** menu.

Groups can be grouped recursively.

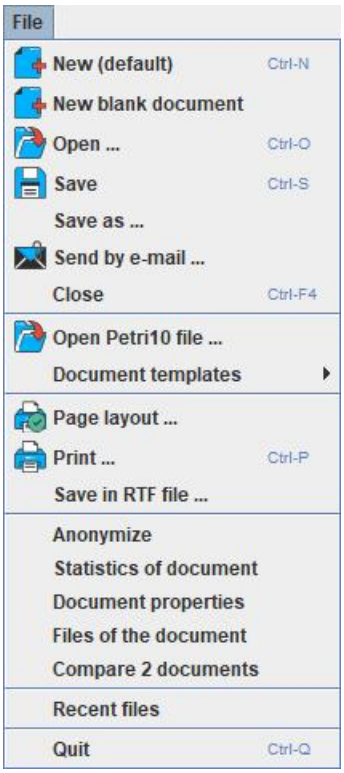
La combinaison **CTRL + F** permet d'effectuer une recherche dans les groupes. Une fois la liste des différents groupes affichée, il est possible de les filtrer ou d'effectuer une recherche dessus.



3. Menus presentation

3.1. File

The menu **File** contains the basics commands : open, close, save, print, etc.



The functionality **New (default)** opens a new document, which will be initialized from the default module's model. You can change the default's model, see Section 14.1, “Document template”

The functionality **New blank document** creates a new blank document.

The functionality **Open** opens an existing document.

The functionality **Save** saves the current document into a file. The default proposed location for the backup is {répertoire home de l'utilisateur}/GRIF/2022/Petri12

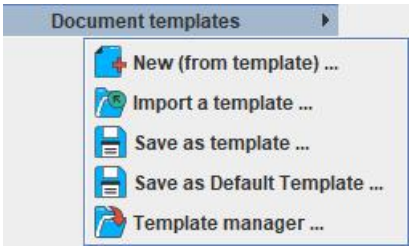
The functionality **Save as ...** lets you save a copy of the file you are working on, with a different name or a different location.

The functionality **Send by e-mail** allows you to attach the current document to an e-mail and then to send it. The configuration of the messaging tool is to be done in the application options Section 16.2, “Executables”

The functionality **Close** lets you close the current document. A window offers to save the file if changes have been made.

Function **Open Petri10 file...** allows the opening of old files in Petri10 format.

The menu **Document templates** includes features related to document reuse and pre-configuration, see Section 14.1, “Document template”.



The print functions **Page layout**, **Print** and **Save in RTF file** are described in the section Section 15, “Printing”

The functionality **Anonymize** deletes all the comments and names filled in by the user. The document does not contain any information helping to understand it.

The functionality **Statistics of document** allows to have some information about the document (number of pages, number of groups, etc.).

The functionality **Document properties** allows you to edit the properties of the current document. The fields include: name, creation date, creator, description, version, ... This function is described more specifically in the section Section 4.14, “Document properties / Track change / Images management”

The functionality **Files of the document** includes files within the current document. These files can then be exported in your reports. This feature is described more specifically in the section Section 4.16, “Files of the documents”.

The functionality **Compare 2 documents** highlights the changes made between 2 versions of the same document. This feature is described more specifically in the section Section 4.15, “Compare 2 documents”.

The menu section **Recent files** list recently opened files to access them faster.

The functionality **Quit** exits the application. Open documents will be closed.

3.2. Edit

The menu **Edition** contains all the commands needed to edit the current model.

The functionalities **Undo** and **Redo** allow you to cancel or redo the last actions performed. The size of the history of undoable actions are configurable in the application options.

The functionalities **Copy**, **Cut**, **Paste** and **Paste and renumber** are described more specifically in this section Section 4.1, “Copy / Paste / Renumber (without shortcut)”

The functionality **Remove** deletes selected graphic elements.

The functionalities **Overall change ...** and **Selection change ...** search and then replace names and identifiers of the current document or selection. This feature is described more specifically in the section Section 4.3, “Overall change” and Section 4.4, “Selection change”.

Actions **Glue** and **UnGlue** link or unlink graphical objects between them. This allows to fix the position of objects. This feature is described more specifically in the section Section 4.10, “Gluing/Associating graphics”

The menu **Automatic layout** gives access to different modes of graphical layouts. This layout can be configured in the document options. See Section 16.3, “Graphics”

The functionality **Select all** selects all the graphical elements of the page.

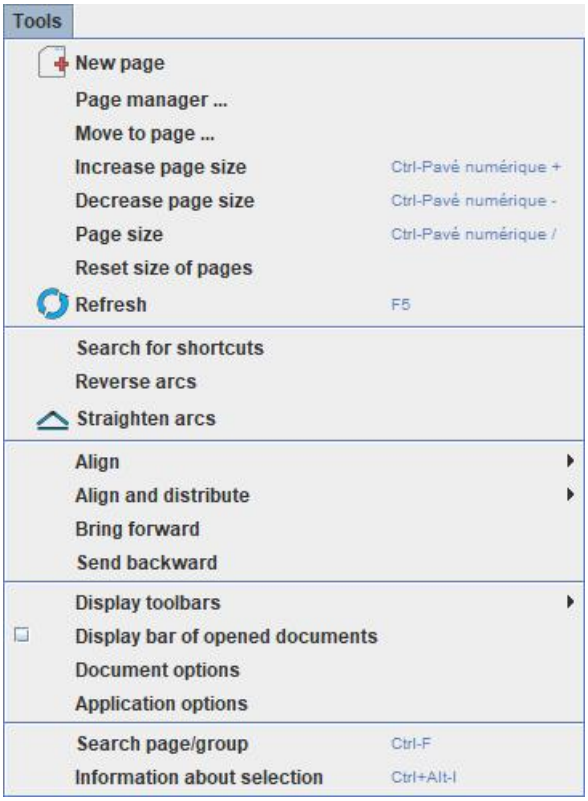
The functionality **Clear selection** deselects items in the current selection. The selection is then blank.

The functionality **Select connected part** selects all graphical elements connected to each other by a link. This feature is described more specifically in the section Section 4.7, “Selecting connex (adjacent) parts”

The functionality **Properties** edits the logical properties of the current selection.

3.3. Tools

The menu **Tools** contains all the commands needed to manage the current model (management of pages, alignments, options ...).



The functionality **New page** : Create a new graphical page on the current document.

The functionality **Page manager ...** : Open a page manager on which you can rearrange the pages of the document.

The functionality **Move to page ...** : Move the current selection to another page or group in the document.

The functionality **Increase page size** : Increase the graphical input area of the current page.

The functionality **Decrease page size** : Decrease the graphical input area of the current page.

The functionality **Page size** : Open a window to manually configure the size and zoom of the current page. This feature is described more specifically in the section Section 4.8, “Zoom and page size”

The functionality **Reset size of pages** : All pages will be reset to factory format.

The functionality **Refresh** : Refresh the graphical objects in the current page.

The functionality **Search for shortcuts** : Opens a window listing the element's references and lead toward these references. This feature is described more specifically in the section Section 2.4.1, “Search shorcuts”

Function **Reverse arcs** allows to reverse the direction of the arc between the place and the transition.

Function **Straighten arcs** allows to eliminate the breaks of the arcs in order to obtain a rectilinear arc.

The menu **Align** provides alignment functions for graphical objects. These features are described more specifically in the section Section 4.5, “Alignment”.

The menu **Align and distribute** contains the same functions as in the menu **Align**. In addition, it is possible to distribute, in the enclosing space, the elements of the selection.These features are described more specifically in the section Section 4.5, “Alignment”.

The functionality **Bring forward** : Move the selected elements one layer forward.

The functionality **Send backward** : Move the selected elements one layer backward

The **Display toolbars** menu lets you show or hide certain shortcut groups from the toolbar.

The functionality **Display bar of opened documents** : Displays in the lower part of the application, a shortcut bar to access documents already opened in GRIF.

The functionality **Document options** : Opens a window to configure the document options. You have the possibility to configure a very large number of GRIF-Workshop's features (cf. Section 16, “Options of GRIF - Petri Nets with predicates”). Some options only apply to the application and are accessible via the menu **Application options**, and others are relative to the document being edited and are defined in the menu **Document options**. However, to avoid having to redefine your options between each document, document options are also available in the application options.
These options will then be applied to all newly created documents.
You can also save the current document settings as the default settings for the application. To do this, open the window **Application options**, then the tabulation **Options** and finally check **Save the options of the current document as default options in the application**.
You will find in this same panel the possibility to override the document options by the application options. To do this, check **The application manages the default options of the documents. Apply the default options to the current document**.

The functionality **Application options** : Opens a window to configure the application options. This window is described more specifically in the section Section 16, “Options of GRIF - Petri Nets with predicates”

The functionality **Search page/group** : Find and locate a group or document page.

The functionality **Information about selection** : Display a window based on the selected graphical elements. This window gives additional information about the current selection.

3.4. Document

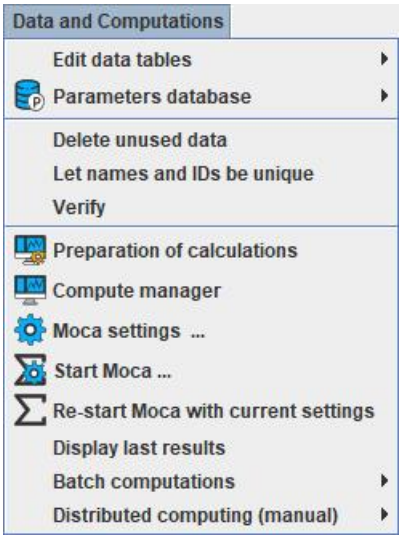
The menu **Document** gives access to all documents being modified or produced.

The functionality **Next** : Selects the next document

The functionality **Previous** : Selects the previous document

3.5. Data and Computations

The menu **Data and computations** is divided into two parts : data management (creation and management of the different parameters) and the parameterization / launch of the calculations (calculation duration, sought calculations ...).



The menu **Edit data tables** provides access to a non-blocking window set that presents the data as tables.

The menu **Parameters database** groups all the functionalities concerning the connection of the application to a specific parameters' database. For more details on the parameters databases, refer to Section 13, “Database of parameters”.

The functionality **Delete unused data** : Cleans the document by deleting unused data. A window opens and proposes to manually select the elements to delete.

The functionality **Let names and IDs be unique** : Identifies and modifies duplicate data in the model. In normal use of the software, it is not necessary to use this function.

The functionality **Verify** : Checks model data and displays errors.

Function **Preparation of calculations** is a utility that lets you run multiple calculations in parallel on one or more documents at a time. This function is generally used to perform sensitivity analyzes. For more information on using this tool, refer to Section 11.5, “Preparation of calculations”.

The functionality **Compute manager** : Opens a non-blocking window to manage the calculations launched by the application. For more details on the compute manager, refer to Section 11.2, “Compute manager”.

Function **Moca settings...** opens the calculation configuration window. For more information on setting calculations, see Section 11.1.1, “Configuring the computations”.

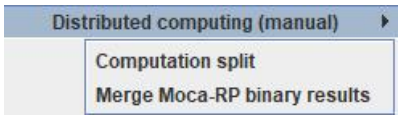
The functionality **Start Moca ...** : Opens the configuration window for Moca computations. For more details on Moca computations, refer to Section 11.1.1, “Configuring the computations”.

The functionality **Re-start Moca with current settings** : Restarts the computation with the last parameterization of Moca computations.

Function **Display last results** : Displays the result window of the last calculation performed. For more information on the result window, refer to Section 11.1.2, “Reading the results”.

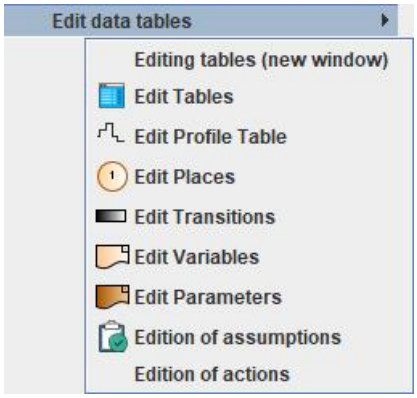
The menu **Batch computations** includes the functionalities of multiple computations by variation of parameters. For more details on batch computation, refer to Section 11.4, “Batch computation”.

Function **Distributed computing (manual)** allows you to split the calculations, so that it can be run in parallel on several separate machines.



3.5.1. Edit data tables

The menu **Edit data tables** provides access to a non-blocking window set that presents the data as tables.



Editing tables (new window) : Opens a new non-blocking window containing all the editing tables of the data.

Edit Tables : Opens a non-blocking window containing the Tables editing panel. Tables settings are detailed here Section 8, “Profile tables”.

Edit Profile Table : Opens a non-blocking window containing the profile tables editing panel. Profile tables settings are detailed here Section 8, “Profile tables”.

Edit Places : Opens a non-blocking window containing the places editing panel. Places settings are detailed here Section 2.2.1, “Configuring the Places”.

Edit Transitions: Opens a non-blocking window containing the transitions editing panel. Transitions settings are detailed here Section 2.2.3, “Configuring the Transitions”.

Edit Variables : Opens a non-blocking window containing the variables editing panel. Variables creation and settings are detailed here Section 9, “Statistics and Setup of Variables”.

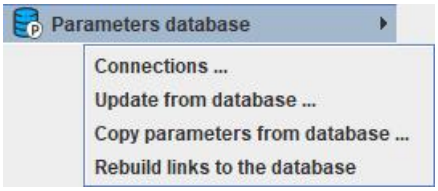
Edit Parameters : Opens a new non-blocking window containing the editing table of parameters.

Edition of assumptions : Opens a new non-blocking window containing the editing table of assumptions. The assumptions' settings are detailed here Section 4.17, “Hypothesis”.

Edition of actions : Opens a new non-blocking window containing the editing table of actions. The actions' settings are detailed here.

3.5.2. Parameters database

The menu **Parameters database** groups all the functionalities concerning the connection of the application to a specific parameters' database. For more details on the parameters databases, refer to Section 13, “Database of parameters”.



The functionality **Connections ...** : Opens the parameter connection's manager.

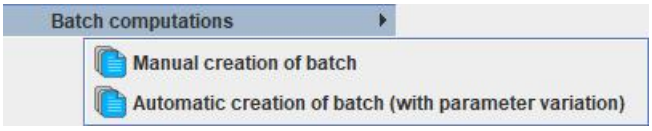
The functionality **Update from database ...** : Updates the settings of the current document parameters that are connected to a database by updating their values. Opens a window to select the data to update.

The functionality **Copy parameters from database ...** : Imports from a parameter database a set of data in the current document. Displays a database parameter table, the user can select the data to import into his document.

The functionality **Rebuild links to the database** : Attempts to reconnect parameter's settings of a document to data from the database Opens a window that highlights parameters that can be reconnected.

3.5.3. Batch computations

The menu **Batch computations** includes the functionalities of multiple computations by variation of parameters. For more details on batch computation, refer to Section 11.4, “Batch computation”.



The functionality **Manual creation batch** : Opens a window to create calculation batches. Each batch must be configured by the user.

The functionality **Automatic creation of batch (with parameter variation)** : The user fills in the parameters to be varied, the application generates the calculation batches by making all the combinations of the parameters.

3.6. Simulation

Menu **Simulation** includes all the features available for interactive simulation. For more information on interactive simulation, see Section 10, “Simulation interactive”.

Function **Start simulation** : start the interactive simulation.

Function **Go to initial state** : restart the simulation

Function **Go back (one step)** : allows you to step back one step in the simulation history

Function **Go forward (one step)** : replay the last step in the simulation history

Function **Go next step** : allows to play the next transition present in the scheduler of the transitions of the simulation.

Function **Stop simulation** : allows to stop the current simulation and to return to an editable model.

Function **Display simulator** : opens a non-blocking window displaying interactive simulation tools.

Function **Legend** : displays a legend about the color codes used by the simulator

3.7. Group

The menu **Group** concerns the capture and management of sub-models grouped into independent subsets.

The functionality **Group** : Puts the selected elements into a new group. A new group graphic object is created. Selected elements are moved within the group.

The functionality **New group** : Create a new empty group on the current page.

The functionality **Name change** : Edits the name of the selected group.

The functionality **Picture change** : Assigns or modifies the graphical rendering of the group, by adding an image. The name of the group will be displayed below the image.

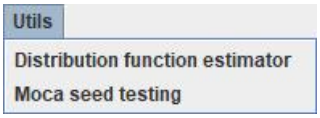
The functionality **Ungroup** : Removes the group and creates all the elements that the group used to contain.

The functionality **Edit group** : Open the group page.

The functionality **Quit group edition groupe** : Go back to the parent page of the group.

3.8. Utils

null

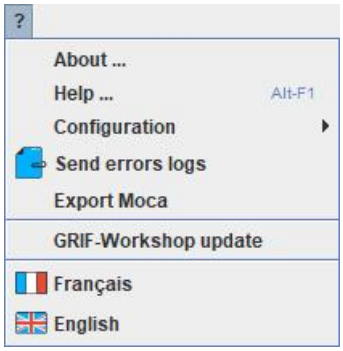


The functionality **Distribution function estimator** : Estimator of a distribution function (Hazard Rate) with a coefficient

Function **Moca seed testing** opens a utility to validate the seeds used by the calculation engine.

3.9. ?

The menu **?** combines several GRIF global configuration functions and provides access to the module's online help.



The functionality **About ...** : Opens an information window about the software version used.

The functionality **Help ...** : Provides access to the module's online help.

The menu **Configuration** groups together several configuration elements of GRIF.

The functionality **Send errors logs** : Sends an email to your reseller with the module's log files.

Function **Export Moca** : Generates the .mk13 file from the current model. This file is generated by GRIF for the Moca-RP calculation engine.

The functionality **GRIF-Workshop update** : Updates GRIF. This function detects the existence of a more recent version of GRIF. If such a version exists, you will be offered to install it.

The functionality **Français** : Change the application language to French.

The functionality **English** : Change the application language to English.

3.9.1. Configuration

The menu **Configuration** groups together several configuration elements of GRIF.

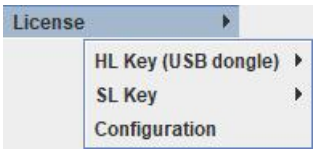
The menu **Licence** groups the configuration functions of the license server. For more information on the use of licenses, please refer to the GRIF installation manual.

The menu **Associate GRIF files** forces your operating system to associate the GRIF files and the different modules that open them.

The menu **Network configuration** : Configures network access to update the system.

3.9.1.1. License

The menu **Licence** groups the configuration functions of the license server. For more information on the use of licenses, please refer to the GRIF installation manual.



The menu **Hardware Licence (HL)** configures USB license dongles.

The menu **Software Licence (SL)** configures license servers that do not require a USB dongle.

The functionality **Configuration**: Configures the access to the license server.

3.9.1.1.1. HL Key (USB dongle)

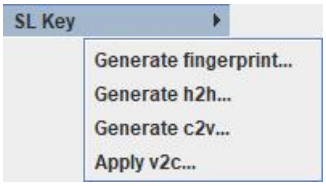
The menu **Hardware Licence (HL)** configures USB license dongles.

The functionality **Generate c2v...**: Generates a c2v (Client To Vendor) file. This file will be requested by your reseller to create an update of your license.

The functionality **Apply v2c...**: Applies a v2c (Vendor To Client) file. This file will be returned by your reseller to apply the update of your license.

3.9.1.1.2. SL Key

The menu **Software Licence (SL)** configures license servers that do not require a USB dongle.



The functionality **Generate fingerprint...**: Generates a c2v (Client To Vendor) file. This file will be requested by your reseller to create your license.

The functionality **Generate h2h...**: Generates a h2h file (Host To Host)This file is needed to transfer your license to a new server. This feature must be used on the source server. For more information on the license transfer procedure, please refer to the GRIF installation manual.

The functionality **Generate c2v...**: Generates a c2v (Client To Vendor) file. This file will be requested by your reseller to create an update of your license.

The functionality **Apply v2c...**: Applies a v2c (Vendor To Client) file. This file will be returned by your reseller to apply the update of your license.

3.9.1.2. Associate GRIF files

The menu **Associate GRIF files** forces your operating system to associate the GRIF files and the different modules that open them.

The functionality **For current user** : Associates GRIF files to the current user

The functionality **For all users** : Associates GRIF files to every users. This operation requires administrator rights.

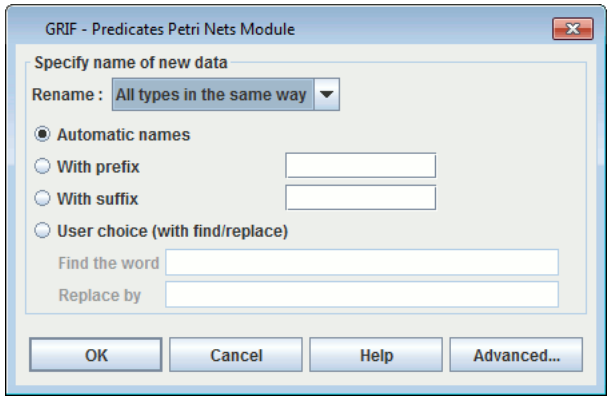
4. Data Entry Aids

To simplify model creation the Petri Nets with predicates module has different data entry aids to automate time-consuming operations.

4.1. Copy / Paste / Renumber (without shortcut)

To assist with the entry of the repeated parts of the Petri Nets "Copy / Paste and Renumber" mechanisms have been provided. This operation is carried out in 6 steps:

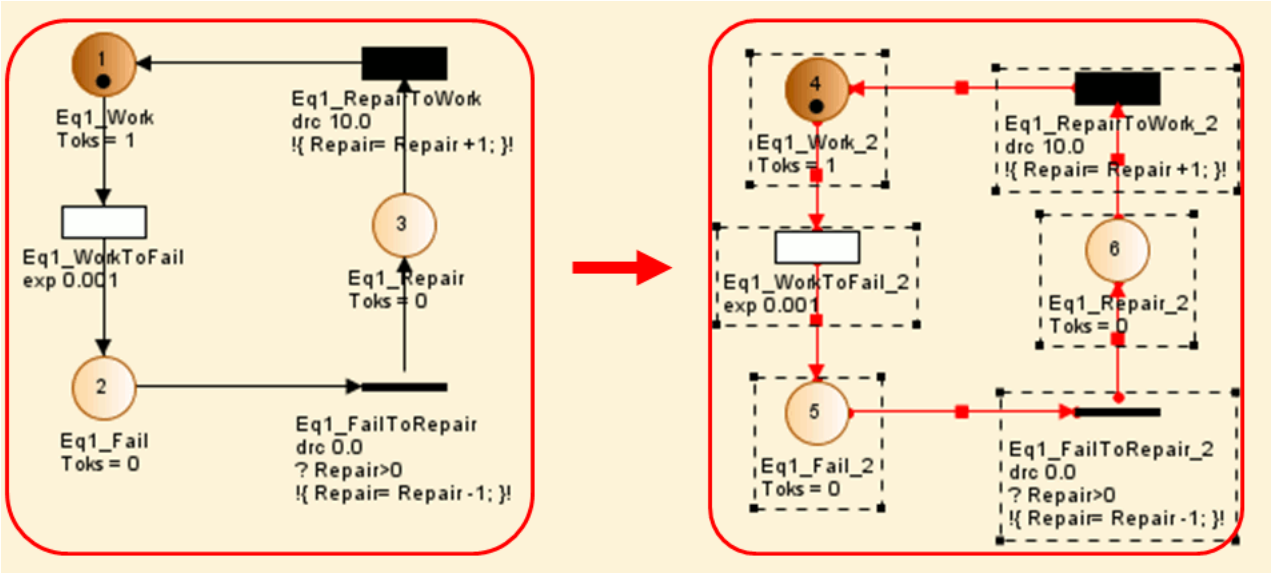
1. Select the part to be copied.
2. Click the **Copy** icon, or use menu **Edit - Copy** or the shortcut Ctrl + C.
3. Click the **Paste and Renumber** icon, or use menu **Edit - Paste and Renumber** or the shortcut Ctrl + R.
4. A window appears where you choose the way to rename the elements.



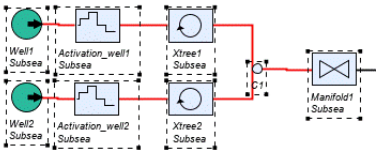
Automatic names choose allows to add a number of the name of the place.

5. The previously selected part is copied and the copy is selected.
6. Move the copy to the desired location.

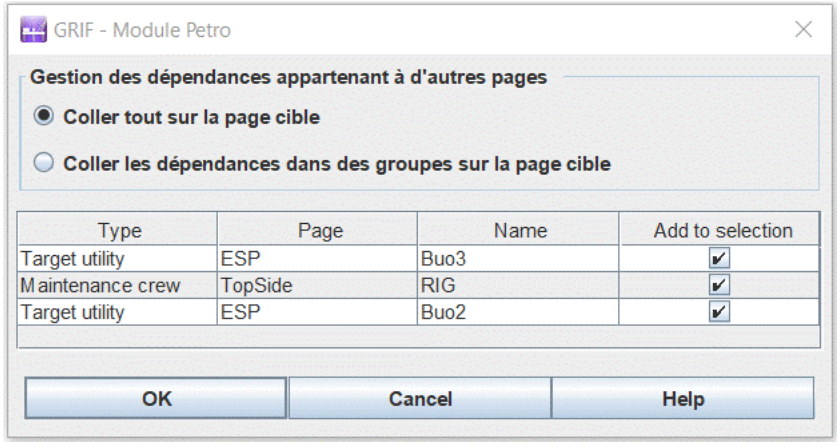
We then obtain the net shown in the figure opposite/ below: Places 1, 2 and 3 of the original have been transformed into 4, 5 and 6 for the copy.



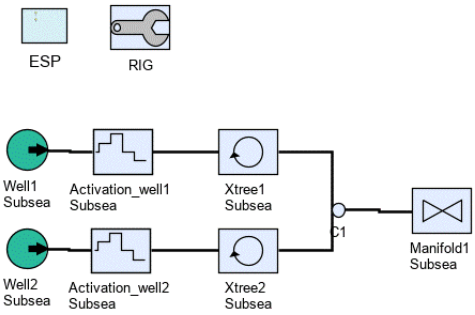
When copying to a new document, There may be dependencies with the copied data.



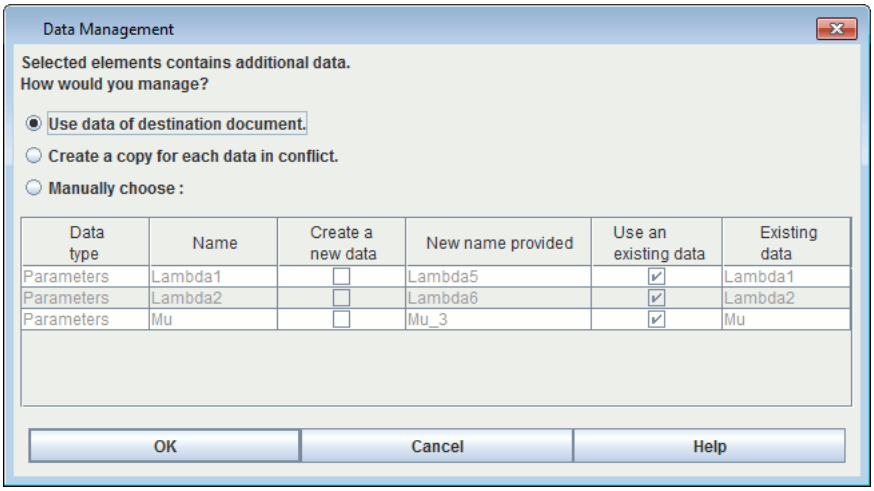
The following window opens and enables to manage indetified dependencies by GRIF :



After managing of dependencies, these are pasted with the selected data



When copying to a new document, any data conflicts are handled in the following window:

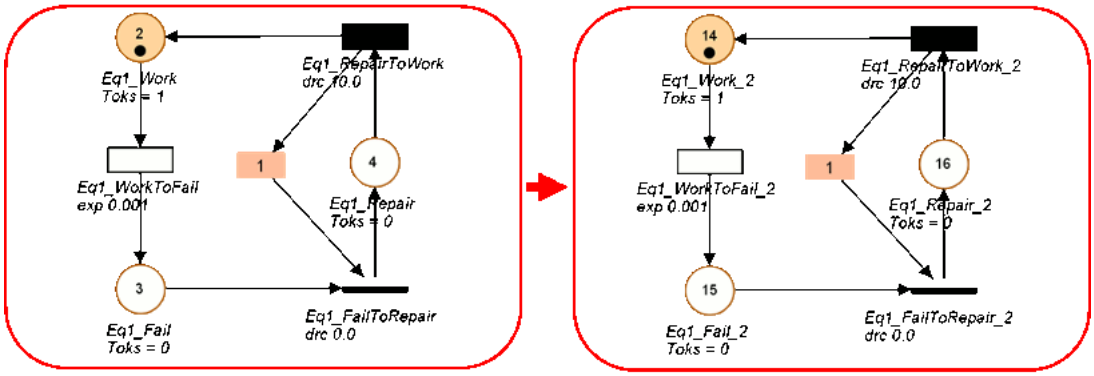


This window shows all the data which has the same name in the source document and the destination document. There are three choices:

1. Use data of destination document, this will replace the occurrences of the data in the source document by the data with the same name in the destination document.
2. Create a copy for each data in conflict, this will replace the occurrences of the data in the source document by a copy with a name with the suffix "copy".
3. Manually manage conflict, this allows you to choose whether you use the existing data or not, depending on the data. You can also specify the name of the copy by double clicking on the box in the "destination document"

column. The names in this column are normally masked when the **Use existing** check box is selected, since it is the data which is already in the destination document which will be used.

If the selected part is made up with a shortcut, the shortcut refers always at the same source.



4.2. Ordinary Copy/Paste

In addition to the "Copy / Paste and Renumber" command there is an ordinary "Copy / Paste" function. It is used to make a single copy without renumbering. We thus obtain double elements which, from a formal viewpoint, is incorrect but which must be temporarily tolerated to simplify data entry.

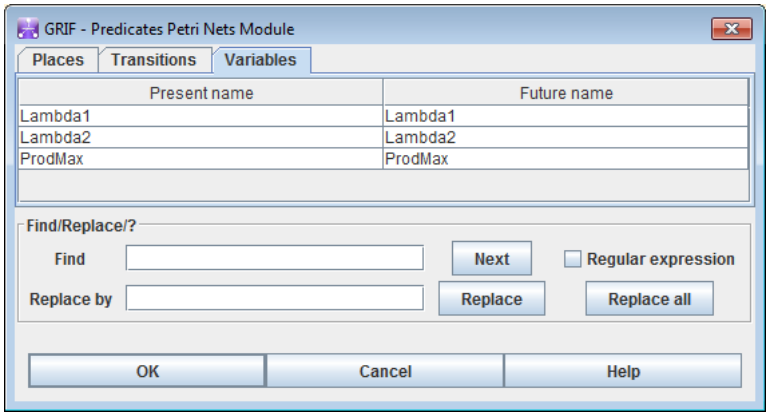
Where possible, the "Copy / Paste and Renumber" function must be used in preference to the simple "Copy / Paste" function to minimise the risk of errors. But when it is used you must take the necessary precautions to re-establish the correct numbering to eliminate the duplicates.

4.3. Overall change

When creating the Petri Nets it may be necessary to change a large part of the elements in the models: changing the names, numbers, etc. The "Replace all" function in the **Edit** menu enables you to perform overall changes:

- Use the **Edit / Overall changes** function.
- Choose the type of elements to be modified among available tabs.
- The "Find / Replace" part changes a character string present in one or more variable labels, place labels or transition labels. It is replaced by the string entered in the "Replace" part.
- The "Renumber" part only concerns the places. It is used to change place numbers. You indicate a **Start** number then specify a constant **Step**, or **Add** a constant value to the current numbers.
- Click **OK** to return to the chart. The changes are validated.

The name changes and renumbering can be done manually if the necessary precautions are taken (avoiding duplicates, etc.). You click the **Future number** or **Future name** column and enter the change. Do not forget to validate it with the "ENTER" key.



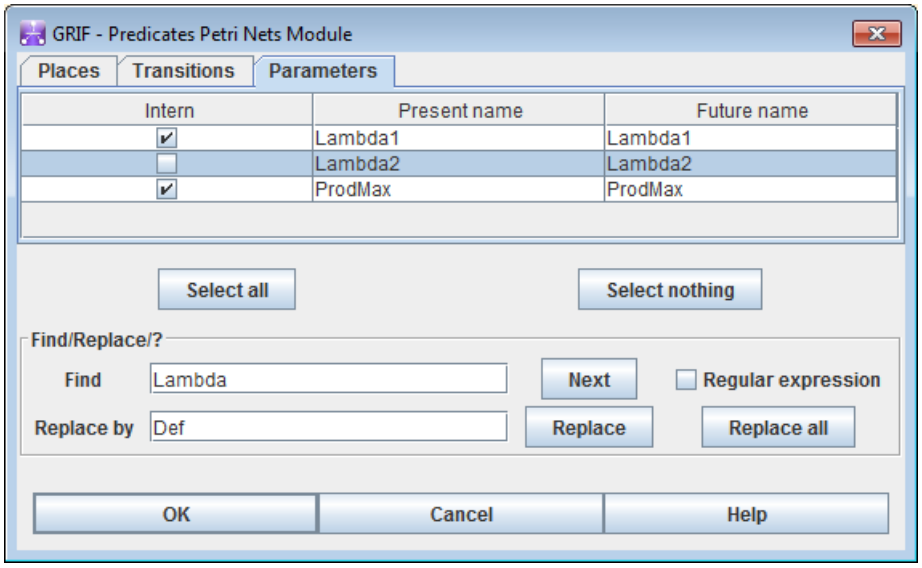
4.4. Selection change

The "Replace selection" function is equivalent to a "Replace all" but only applied to the selected elements.

The only difference is that we must make the distinction between "internal" and "external" variables/parameters.

- "Internal" variable/parameter: only used within the selection
- "External" variable/parameter: used within the selection but also used elsewhere in the model.

Only the "internal" elements can change their name. If a variable or a parameter is recognised as being "external", the check box in the **Internal** column must be selected before it can be modified. The change will only affect the selected part. Everything outside the selection will remain unchanged.



In the above example only the name of parameter "Lambda2" will change (in the "Replace selection" sense) since it is "internal" to the selection. A new parameter called "Def2" (with identical value) will be created and will replace "Lambda2" in the model. The other parameter, which is not "internal", will remain unchanged.

4.5. Alignment

To improve the legibility of the model the selected elements can be aligned vertically or horizontally. To do this, use the **Align** command in the **Tools** menu.

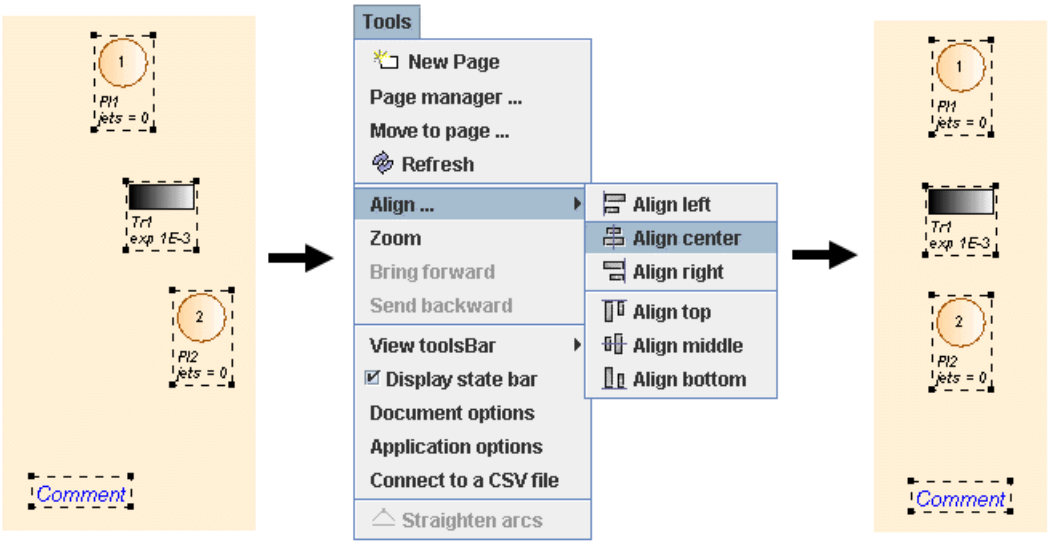
The following figure shows how the command works. For example, to align selected places and transitions vertically, proceed as follows:

1. Select the elements (places, transitions, comments, etc.) to be aligned;

2. Go into the **Tools** menu and select the **Align** function;

3. Choose the type of alignment: **Align center**;

4. Click left on the mouse.



Similarly, to align elements horizontally select the type **Align middle** which aligns the ordinates while keeping the abscissa constant. The principle is the same as that described above.

4.6. Multiple selection

It may sometimes be useful to select several elements located in the four corners of the input zone. To simplify this type of selection click on each of the desired elements one by one while holding down the Shift key on the keyboard.

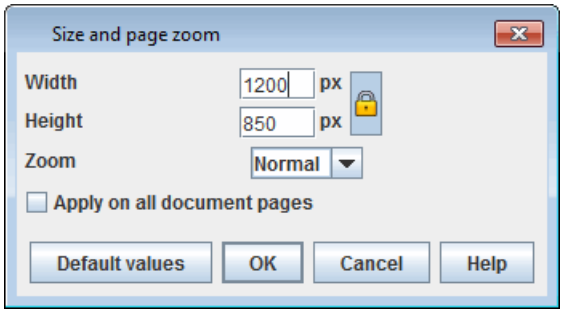
4.7. Selecting connex (adjacent) parts

It is sometimes difficult to select an additional part of a model. To simplify the selection process, select a graphical element then use menu **Select connex part** in the **Edit** menu. The additional part can be selected directly by clicking on the element while keeping the Control button pressed.

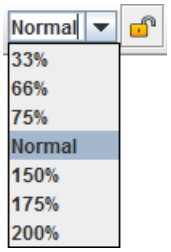
4.8. Zoom and page size

When creating a model, if the page size is not big enough, it can be changed using the menus: **Increase page size** (**Control**+**Keypad +**), **Reduce page size** (**Control**+**Keypad -**), **Page size** (**Control**+**Keypad /**) under the **Tools** menu.

The **Page size** menu enables the user to edit the page dimensions directly.



Page zooms can be modified either by using the toolbar menu:



Or by selecting the display and using **Control**+**mouse wheel scroll up** to zoom or **Control**+**mouse wheel scroll down** to zoom out.

The padlock on the toolbar is used to apply the zoom to the current page or to all pages in the document.

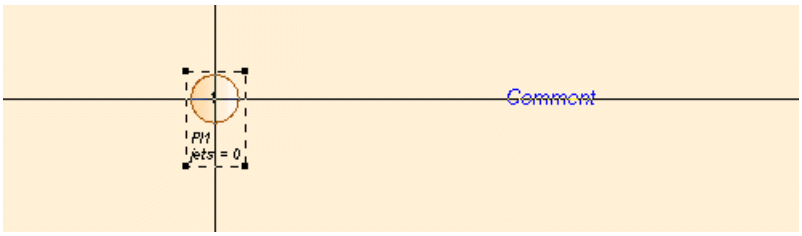
	The zoom applies to all pages in the document.
	The zoom is applied only to the current page.

Note that if an element is selected on the page, the zoom will centre the page on that element.

4.9. Cross hair

To be able to create an ordered and legible model quickly, the **cross hair** can be used to align the different elements with each other (but less accurately than the **Align** function in the Tools menu). The **cross hair** is enabled (or disabled) in the **Graphics** tab of the **Option** menu.

The following picture show how to quickly align two elements of the model.



In order to align horizontally, select **Align au middle** which align keeping constant abscissa.

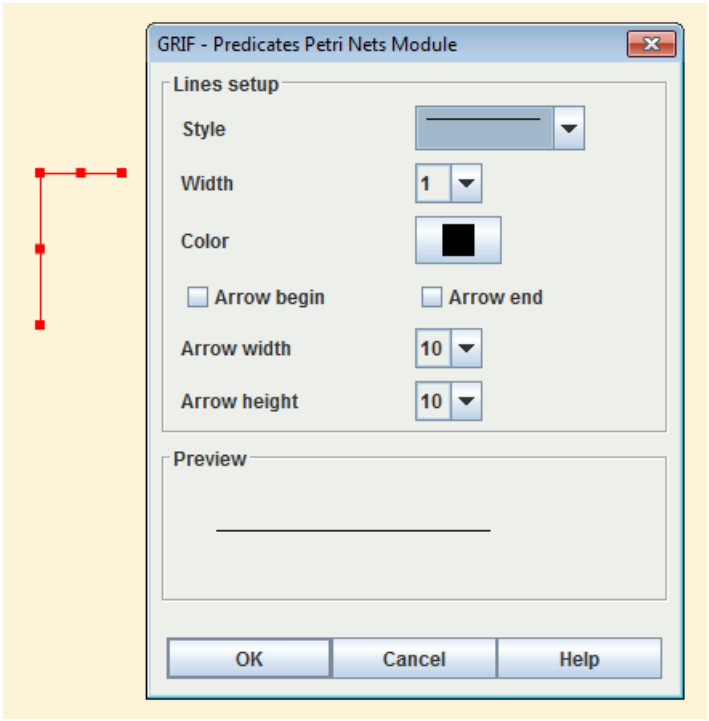
4.10. Gluing/Associating graphics

When objects are where you want, you can glue a set of objects by right-clicking and selecting **Glue**. This command creates a group (a graphical one, not a hierarchical one) with selected objects, so that moving one moves the others.

A double click on an element in the glue group opens the properties window of the element.

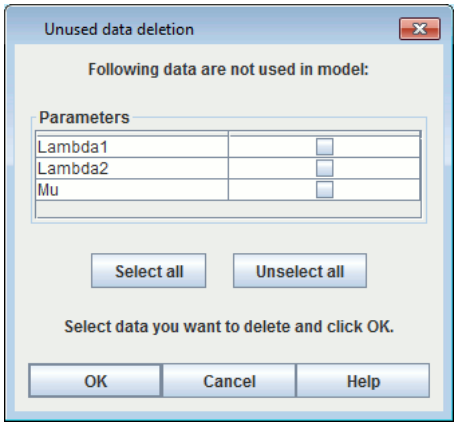
4.11. Line

To be able to draw a line, polyline or arrow, the **Line** can be used. Draw the line and edit properties of line to make an arrow.



4.12. Table Cleaning

Data may not be used anymore, it can be used useful to delete every unused data. To facilitate removal, use **Data and Computations / Unused data deletion** menu.



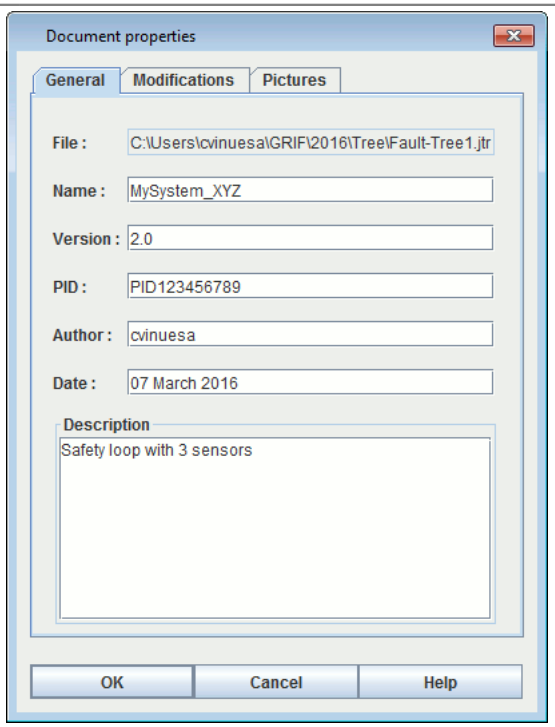
This window displays unused data. Select data you really want to delete and click OK.

4.13. Prototypes

cf. appendix about prototypes

4.14. Document properties / Track change / Images management

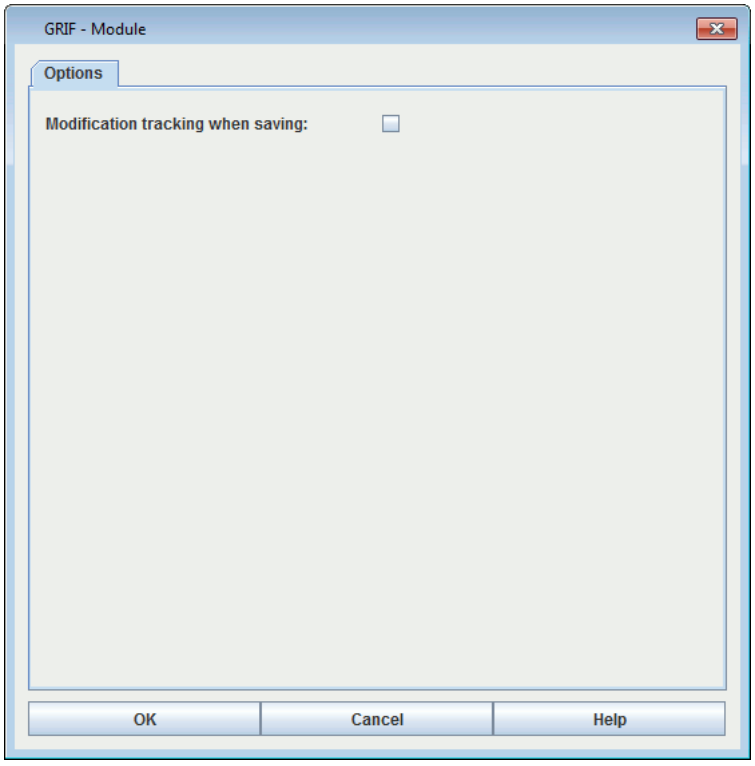
File - Document properties menu enable to save information about document: name, version, comment, ... These information are available in **General** tab.



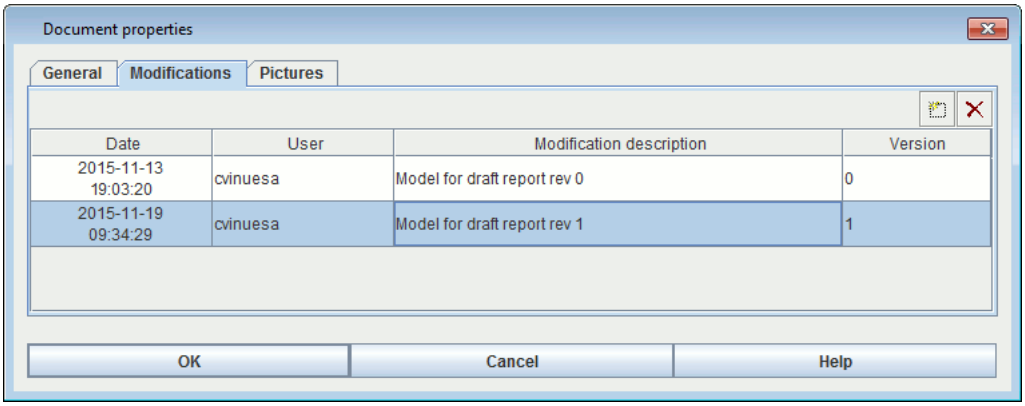
Modification tab enables to save A history of the modifications.

There are two different ways to save modifications:

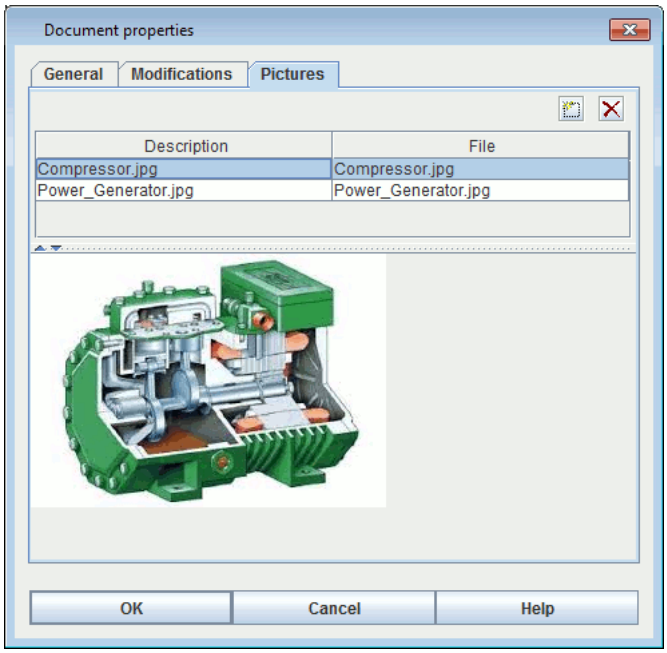
- At each saving by checking: **Modification track when saving** dans **Tools - Document (or Application) options** .



- When the user wants directly in **Modification** tab of the properties using the button



Images may be very useful to represent sub-system. GRIF 2022 enables to save images that can be used in different parts of software (groups, prototypes, ...). Images management is made in **Images** tab.



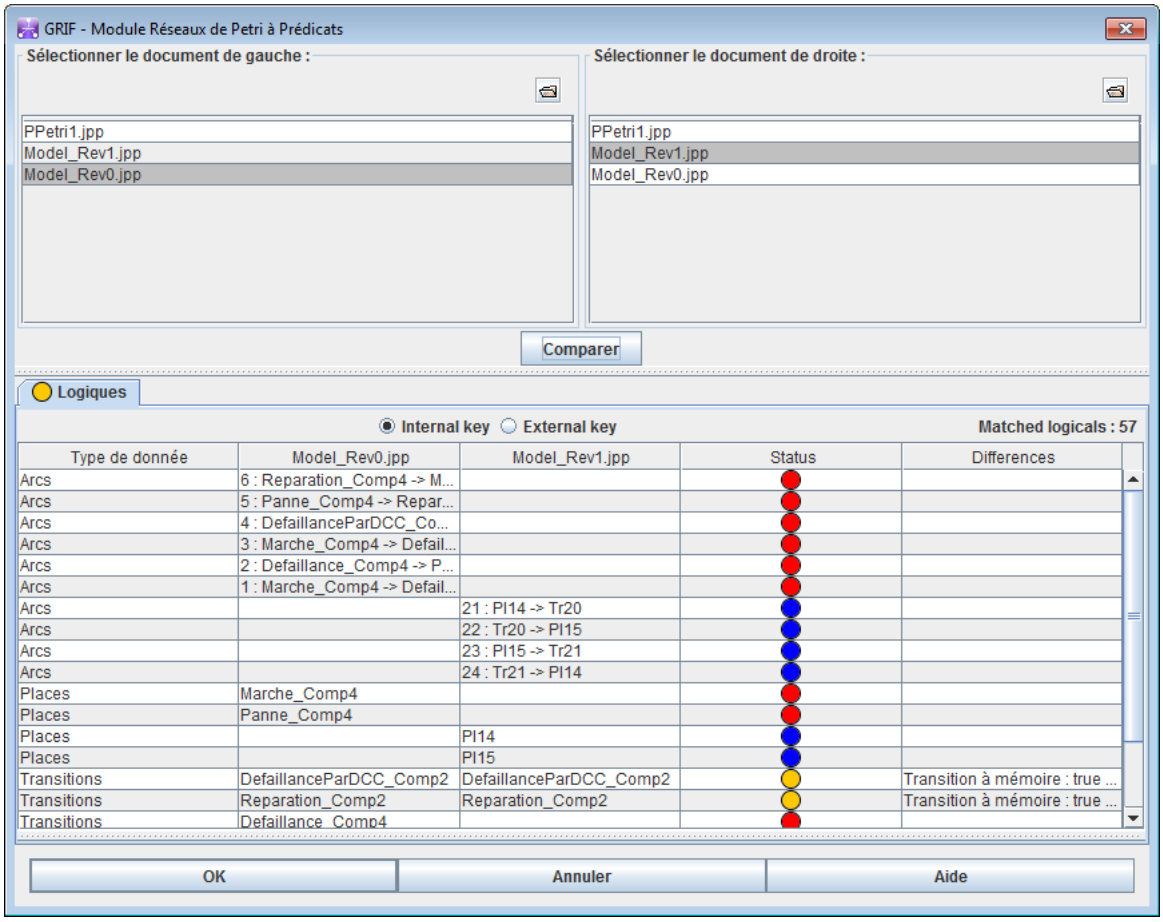
To add a new picture into document, use icon. A double click in **File** column enables to select a picture (jpg, gif or png). A double click in **Description** column enables to give a name or a description to selected image.

Once in document, picture can be linked to a groupe with **Group - Picture change** menu.

Images are saved inside document, pay attention to picture size. Because images are inside document, you have to re-add picture if picture is modified erternaly.

4.15. Compare 2 documents

This function is accessible using **File / Compare 2 documents**. The following window appears:



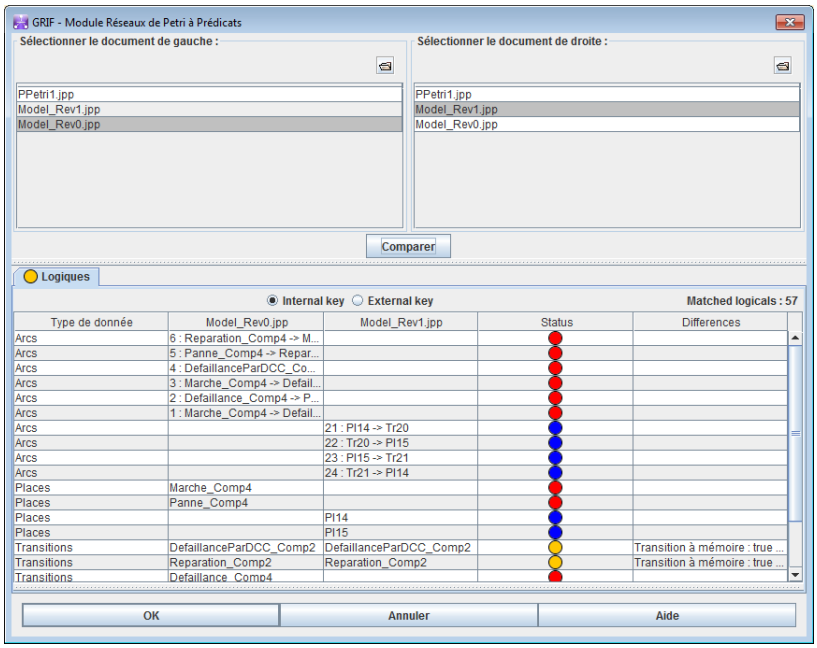
Icon enables loading of the files to be compared.

Click on Compare to launch the comparison.

Difference can be sorted using 3 criteria: internal key, external key or name for nodes

- **Internal key** enumerates the differences according to internal elements of the model for example identifier, creation index, etc...
- **External key** differentiates elements according to the names of the elements of the model.

- **Name for nodes** differentiates nodes according to their names. The external key comparison will be used for others elements.

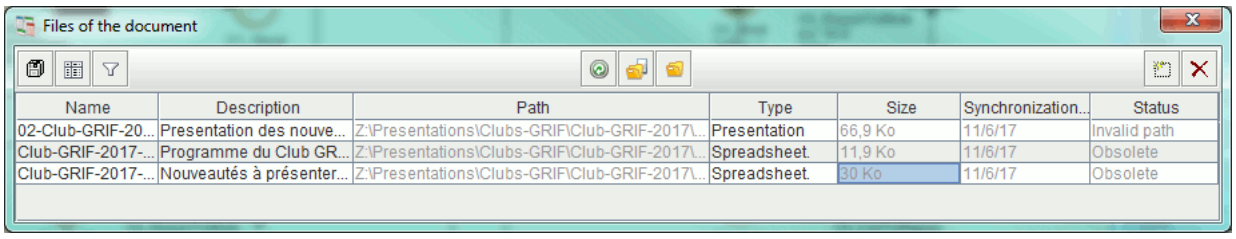


Colour signification is:

- : element is identical;
- : element is added;
- : element is modified;
- : element is deleted.

4.16. Files of the documents

It is possible to associate external file using **File - Files of the document** menu.

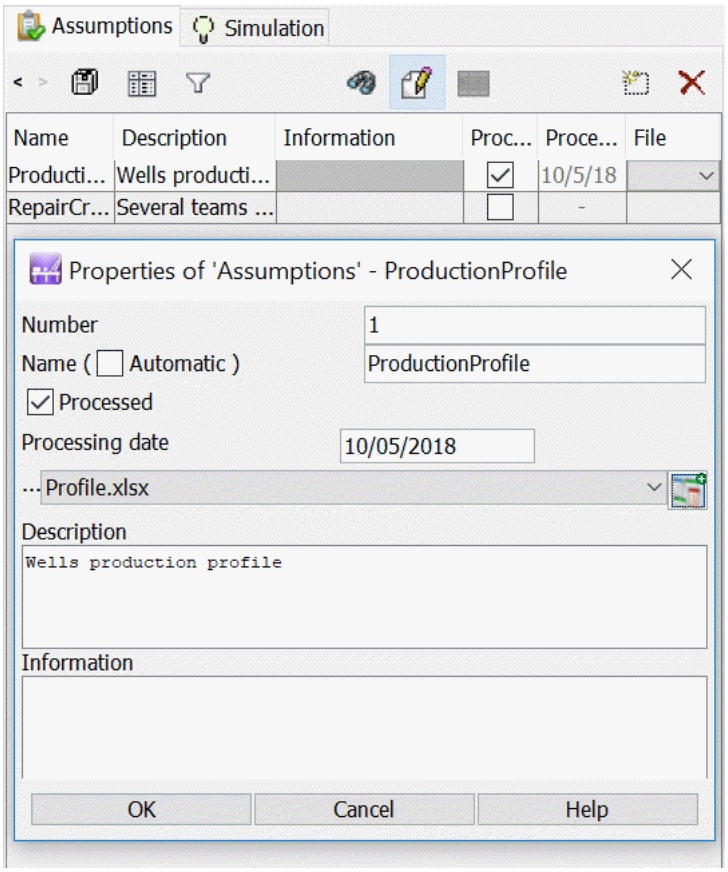


The following icons allow to:

- reload files;
- open files;
- open directory where file is saved.

4.17. Hypothesis

In the data table, in **Hypothesis** tab, it is possible to follow-up and track the studies hypothesis.

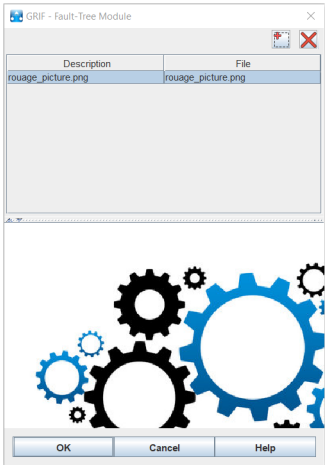


This table enables to take into account the study hypothesis and add file or date to indicate that this hypothesis is taken into account.

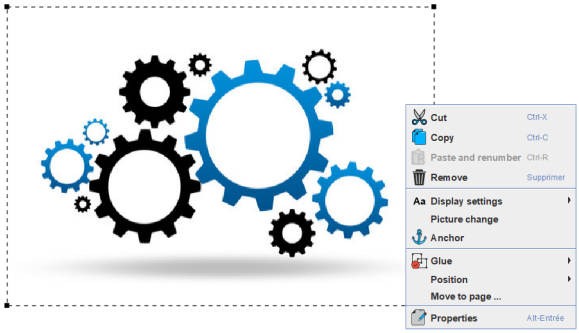
4.18. Picture Anchor

It's possible to anchor a picture in the background. use the action **Picture** of the **Tool barre**.

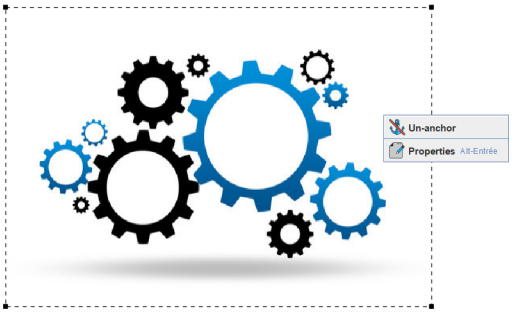
The following screen is display :



It's possible to select a specific picture that is display in background. To anchor the picture, with a right click, select the **Anchor** action. It's possible to change the picture with the **Picture change** action:



To unanchor the picture, with a right click, select the **Un-anchor** action :

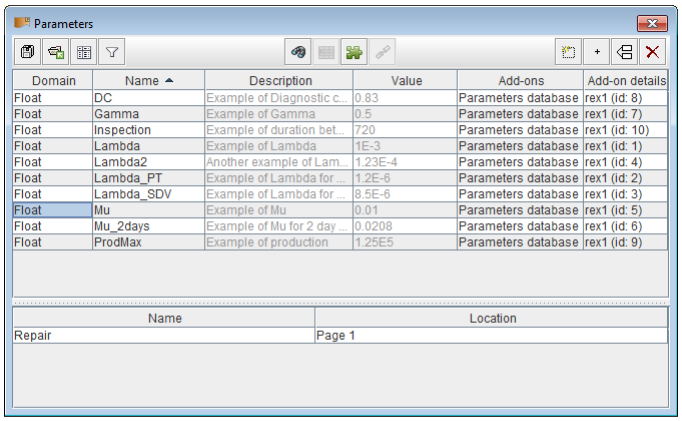


5. The parameters

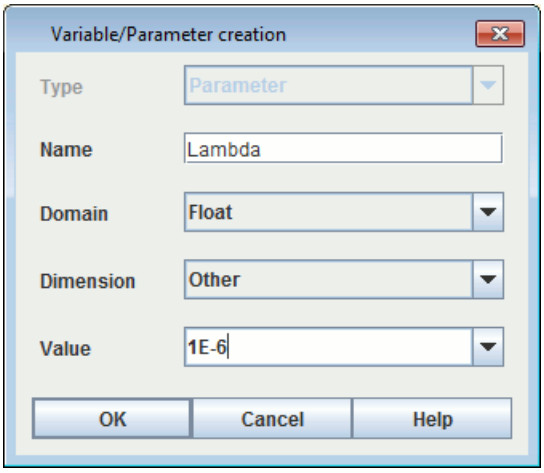
It is possible to create constants which can be booleans, integers or reals. These parameters can then be used for the configuration of different elements of the model (laws, events, transitions, ...)

5.1. Creation

The tab **Parameters** enables the user to define his parameters.



The toolbar enables to do basic operations of the data tables(Section 1, “Description of the Tables”). The button "New" opens the window to create a parameter :



A parameter has a name, a definition domain (Real, Boolean, Integer), a value and a dimension (Failure rate, probability, time, factor, ...) which allow to specify the parameter. This typing is at this moment informative.

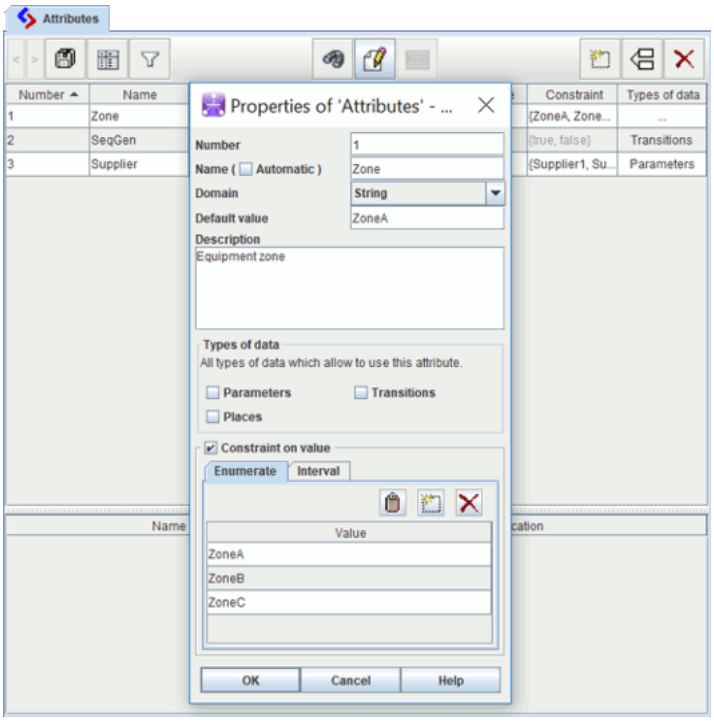
Others additional fields are available in the parameters' table.

Add-On		enables to define the parameter by a GRIF add-on Petri12 is delivered by default with 1 add-on for the parameters : **Parameters database** : is an add-on which enables the user to get the data of his parameter in a database or in a CSV or Excel file. This database is more detailed in this section Section 13, “Database of parameters”.
Add-on details		gives a synthesis of the data defined by the add-on. A double-click on the cell enables the user to modify its definition.
Parameters database	**Database**	Displays the database name containing the parameter.
	Identifier	Displays the identifier of the data in the database.
	Update	Displays the date of the last update of the parameter from the database.

6. Attributes

6.1. Creation

The **attribute** tab enables the user to create attributes that are used to qualify elements defined on system.



The attribute properties are the following ones:

- name;
- domain ;
- default value;
- type of data: to choose where apply the attribute;
- constraint.

The domain type can be of the following values:

- **boolean**: This kind of attribute is a boolean;
- **integer**: This kind of attribute is used to affect an integer value;
- **float**: This kind of attribute is used to affect a float value;
- **string**: This kind of attribute is used to affect a free text.

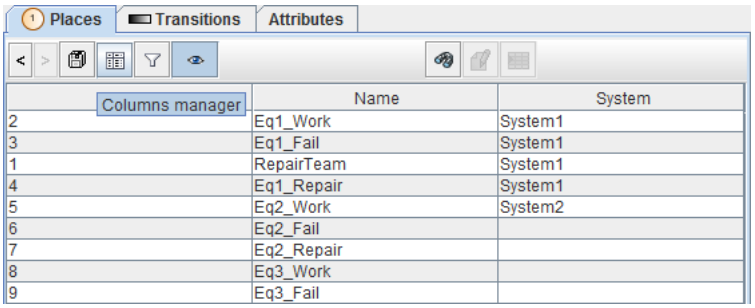
In **Constraint** field, user can enter a constraint on the attribute to ensure the proper use of the attribute in the model.

In addition, the attributes of **float** or **integer** type have a **Constraint type Enumerate** or **Interval**.

6.2. Use of the attributes

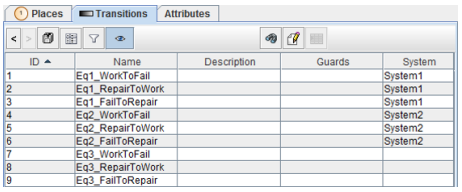
In a Petri net, it is possible to associate attributes:

- Either on places



	Name	System
2	Eq1_Work	System1
3	Eq1_Fail	System1
1	RepairTeam	System1
4	Eq1_Repair	System1
5	Eq2_Work	System2
6	Eq2_Fail	
7	Eq2_Repair	
8	Eq3_Work	
9	Eq3_Fail	

- or on transitions



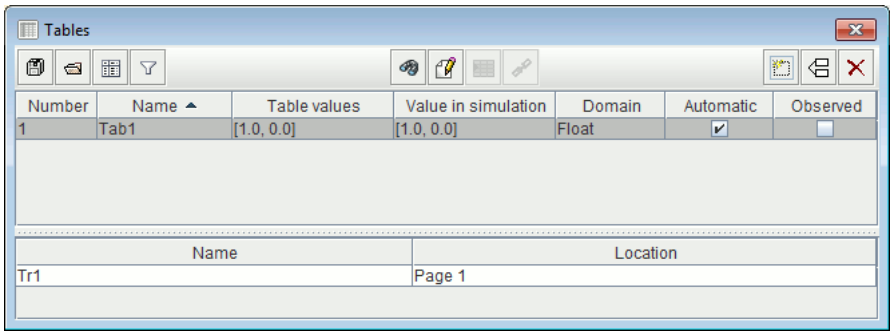
ID	Name	Description	Guards	System
1	Eq1_WorkToFail			System1
2	Eq1_RepairToWork			System1
3	Eq1_FailToRepair			System1
4	Eq2_WorkToFail			System2
5	Eq2_RepairToWork			System2
6	Eq2_FailToRepair			System2
7	Eq3_WorkToFail			
8	Eq3_RepairToWork			
9	Eq3_FailToRepair			

7. Table values

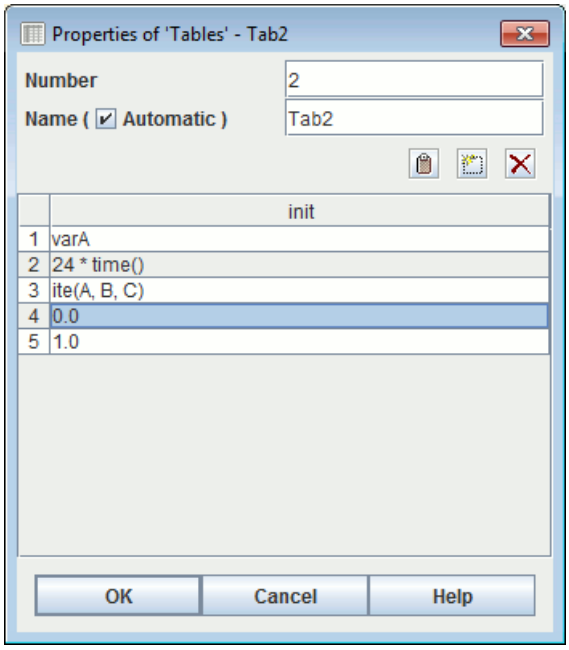
It is possible to create table values. This table have the same behavior as variables, they can be constructed in order to be read in the model, or to store temporary information that will be modified by a transition.

7.1. Creation

Table tab is used to define this table.



Tools bar enables the common data table actions (Section 1, “Description of the Tables”). "Add" button opens a window to create a table:



A table is identified by a number and a name. It contains a list of values that can be numerical or variable expressions.
Others fields are available in the column manager. Some fields are masked by default. To show the others columns see Section 1.2, “Column manager”.

The additional columns accessible are:

Table values	displays value imput by user.
Value in simulation	displays updated values at current simulation time.
Domain	enables to give a domain defintion (Reall, Boolean, Integer) at the table value.
Observed	enables to activate a statitistic computation for each table index. For more information about statistic computation see Section 9, “Statistics and Setup of Variables”.

7.2. Use

The value of a table cell can be changed dynamically during the calculations. To do this, the following syntax can be used in the code of a transition assignment:

Tab1[1] = 2; / Assigns first cell of table Tab1 */*

During simulation, the value of the table can be displayed using a **Dynamic field**:

$data.table.name(Tab2).values

An table can also be used as a function argument:

m = min($Tab1);

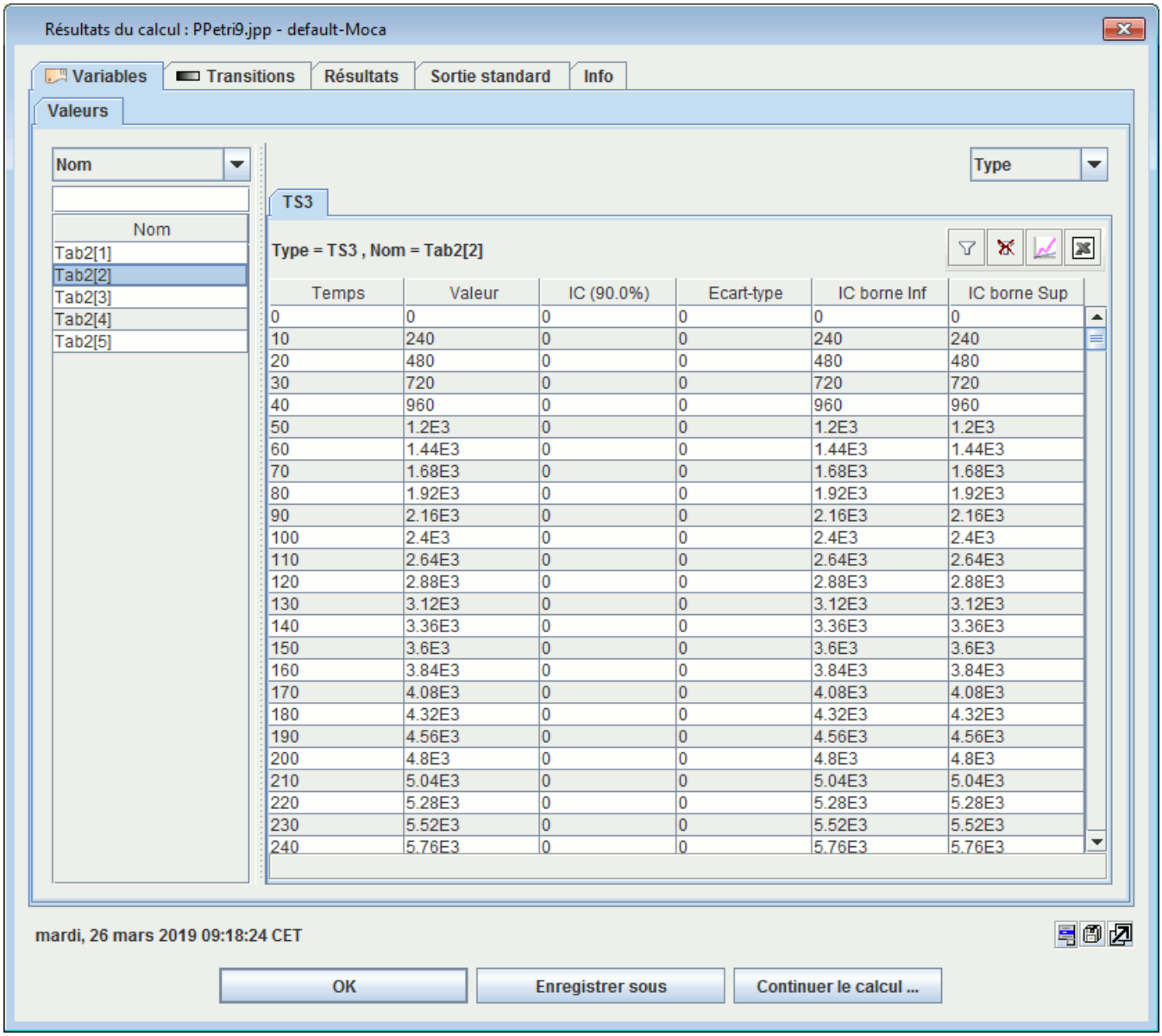
A variable size table can be used. It is the simulation assignments that change the size. This type of table can be used to manage pile or element queues. To use such tables, it is necessary to create them with an empty size (no elements on it).
Assignment example:

```
/* Assigns the 50th cell of the table with the value 1. */
/* The cells not yet initialized [1-49] will take the value 0 */
DynaTab[50] = 1;

/* Increases table size by 1 and affects this cell with value 2 */
DynaTab[DynaTab.size() +1] = 2;
```

7.3. Computations

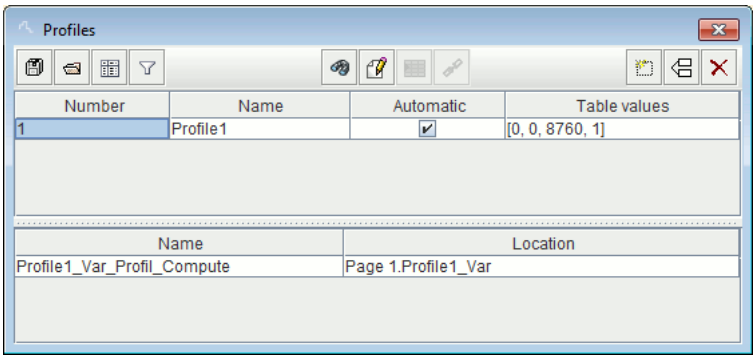
When statistical computations are requested for a table, the statistics are calculated for each index in the table. The results will be displayed in the **Variables** tab of the detailed results window.



8. Profile tables

Profile tables are used to change a variable over time. Variables defined by a profile table are called profiled variables.

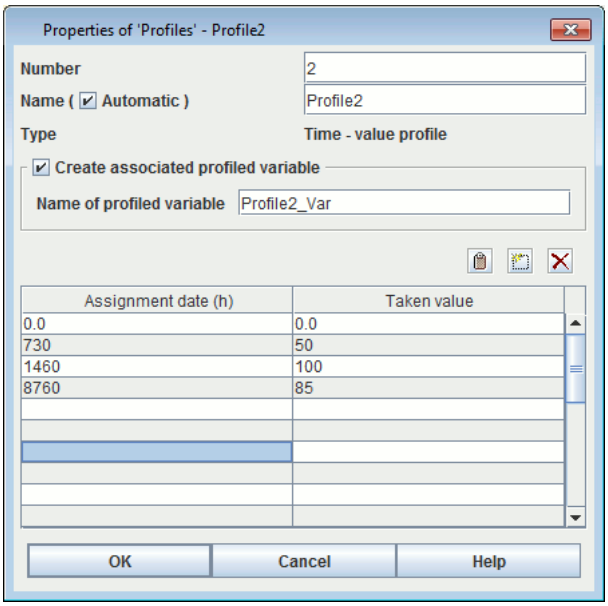
The **Profile** tab enables the user to define his profile tables.



The toolbar enables the classic actions of the data tables (Section 1, “Description of the Tables”). The "New" button opens the profile table creation window.
Several types of definition are possible:

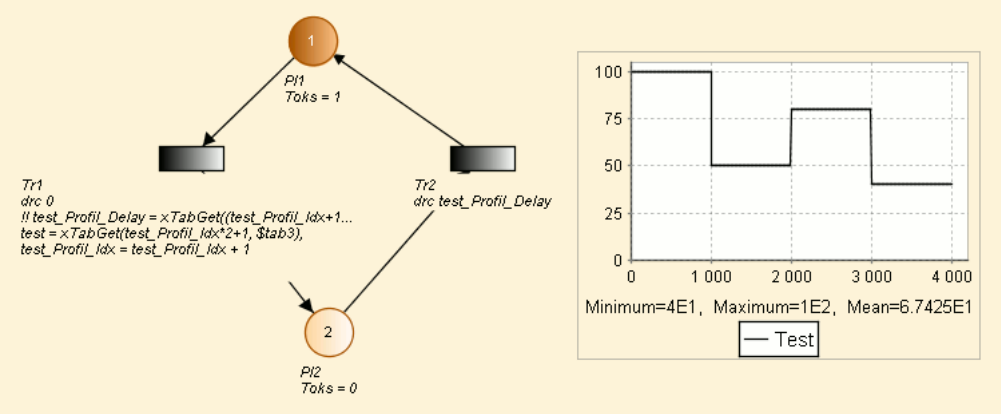
- **Time - value profile** : Table with 2 columns, one for the time and one for the values to be considered at each time.
- **Time - value profile (looped)** : Table with 4 columns: Phase number, Duration, Value and Next phase. Profile entry is done by defining phases. A phase (table row) has a duration in hours and a value that will be taken during this time. Once the time has elapsed, the phase defines as the next phase will be considered.

Below is the Time-Value profile creation window:



A table for "variable profiling" lets you define the modifications of the value of the variable during simulation. In the left column you must enter a date (beginning with 0) and in the right one, enter value that will be assigned to the variable at this date.

For a good profiling, the last date must be greater than history duration. In order to be sure that this date is greater, your are advised to use the following formula: "delay()+1" (wich return history duration + 1).



Profil is not linked to the variable. This is only the Petri Net that has been generated, which is modifying variable according to the profil. That why you must not modify this Petri Net.

On the Time-Value profiles, it is possible to create the profiled variable at the same time as the creation of the table.

It is also possible to create this variable and the associated profiling Petri net by right-clicking on the profile table then:

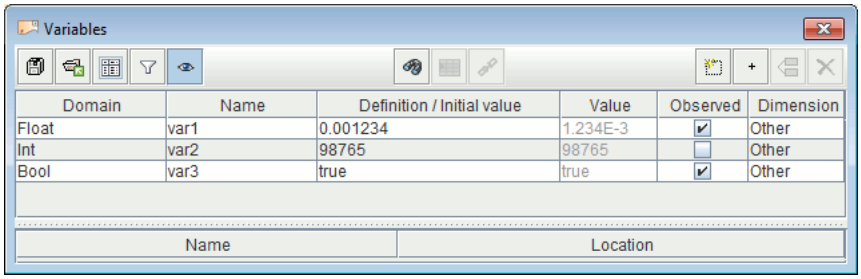
- **Create a profiling net** : Request which variable should be profiled and generates the corresponding profiling network on the current page.
- **Create the profiled variable** : Generates the profiling network and automatically creating the profiled variable.

9. Statistics and Setup of Variables

In addition to the mean marking of the places and of the mean number of times each transition is fired, the simulation can compute a certain number of additional statistics. Statistical results can be obtained on any of the model's variables or combination of variables. To do this, a variable must be declared as "observed". When a variable is "observed", a statistic state (Moca meaning) is created for computation.

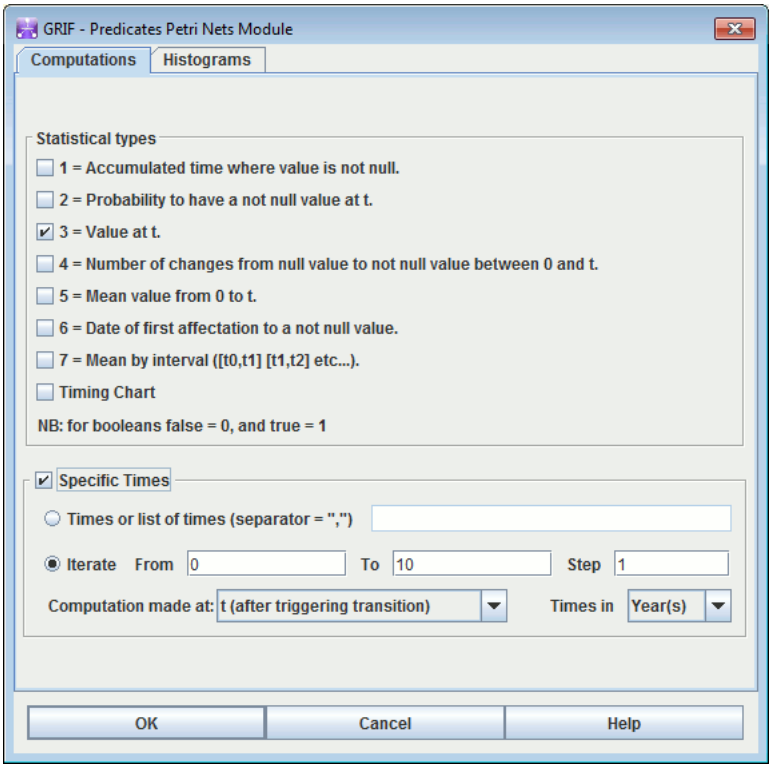
9.1. Definition of statistic states

A statistic state is defined like an "Observed" variable. We must initially define the statistic states we wish to observe. To do this, we have to edit variables from the model: either thanks to the **Data and computations - Edit Variables** menu, or thanks to the **Edit Variables** tab. Then, all we have to do is to set **Observed** property of a variable in order to make it a statistic state.



9.2. Configuration of statistic states (or variables)

Once variable is observed, we have to configure them by specifying types of computations and computation times to be carried out on them, ... To do this, do a right-click on the variable and select **Configuration of computation**.



The window to edit observed variables (statistic states) is made of two tabs. The first tab is for computation, the second one is for histograms. These two tabs enables configuration of **Types of statistics**, **Computation times** and **Histograms**.

9.2.1. Types of statistics

Types of statistical computations are the following:

- **1 - Cumulated time where value is not null:** This is the mean time in which the statistic state is different to 0 on a history.

  Purpose: Mainly used to compute mean availabilities during a history.
- **2 - Probability to have a not null value at t:** This is the probability that the statistic state is different to zero at the end of the history.

  Purpose: Among other things, used to compute the mean availability at the end of a history or compute the reliability (to find if the failure state - which is an absorbent state - is present at the end of a history).
- **3 - Value at t:** This is the mean value of the statistic state at the end of the history.

  Purpose: This type of computation can be used to compute the occurrence of specific event during a history.
- **4 - Number of changes from null value to not null value between 0 and t:** This is the mean number of times during a history that the statistic state changes from a zero value to a non-zero value.

  Purpose: This type of computation can be used to compute the occurrence of specific events during a history.
- **5- Mean value from 0 to t:** This is the mean value of the statistic state on the duration of the history.

  Purpose: Among other things, used to compute the production availability.
- **6- Date of first affectation to a not null value:** This is the mean instant from which the value of the statistic state changes from zero to a value different to zero.

  The "uncensored data" field gives the number of histories for which the simulation has been able to retrieve a value. For this mean statistical result to have a meaning, we must verify that a value has been retrieved for each history (uncensored data = number of simulated histories).

  Purpose: Used to obtain information about the mean instant when a system fails for the first time (reliability computations, estimation of the mean trouble-free operating time, etc.).
- **7- Mean by interval ([t0,t1] [t1,t2] etc ...) :** Expression average value of the state for every time interval

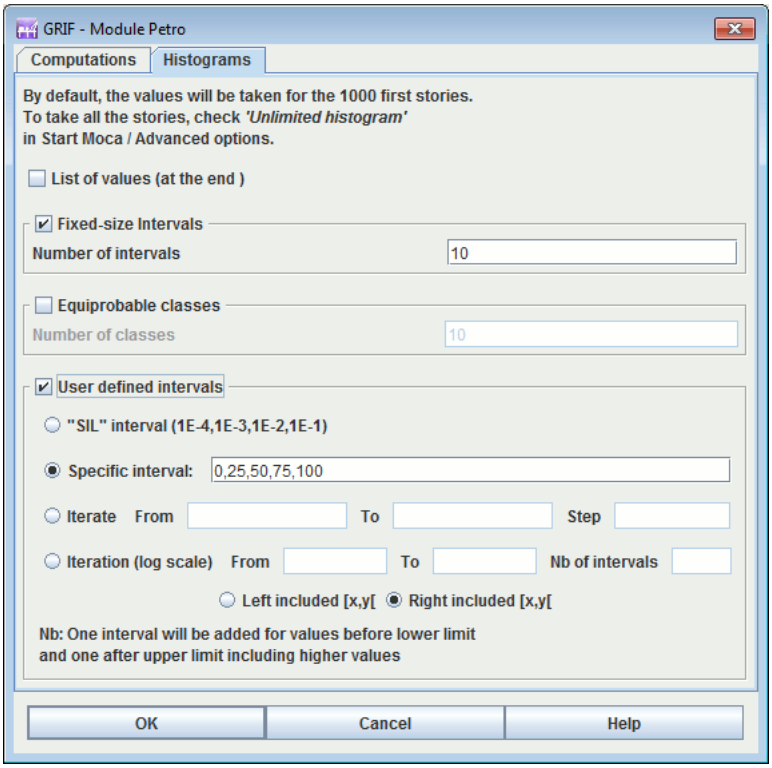
  Was integrated to display the yearly average production of production. Must be used with a list of time (otherwise, the result is identical to a type of statistic 5). The difference between 2 times must be strictly superior in epsilon (a list of including time t and t-epsilon will make impossible the simulation launching).
- **8- Sum value from 0 to t :** represents the sum of the statistic state over history duration.
- **9- Sum by interval ([t0,t1] [t1,t2] etc ...) :** represents the sum of the statistic state between two calculation period.
- **CHRO - Timing chart** Traces the instantaneous value of the variable (as 3) but automatically calculates the "useful" points without taking into account the specified times.

9.2.2. Computation times

Two possibilities are available to define computation times:

- **List of times:** the computations will be performed for the values of t given in this list. Separator is comma.
- **Iterate Form A to B step C:** the computations will be performed for values of t ranging from A to B with a step of C. You can also choose is computations are made before or after transitions triggering.

9.2.3. Histograms



Computations made previously gives mean value for many histories. **Histograms** enable to know how values are distributed during histories. (Cf. Moca User manual for further information)

- **List of values:** provides value at the end of each history.
- **Fixed size intervals:** provides the way value are distributed by cutting intervals of values in X intervals which size is equal.
- **Equiprobable classes intervals:** provides intervals whose probability to contains a value at the end of a story id the same.
- **User-defined Intervals**

  Bounds of intervals can be defined as follows:
  – Automatically defined limits for SIL
  – Manual definition of limits (separate by commas)
  – "Iteration": user choose lower limit and upper limit, and the size of intervals.
  – "Iteration (log scale)": user choose lower limit and upper limit, and the number of intervals. Size of intervals will be computed in order to have same-size-intervals, on a logarithmic scale.

  Moreover, two limits are added at minus infinity, and plus infinity, in order to have a chart containing every history of the simulation.

  When limits are chosen, user has to choose between "left included" or "right included".

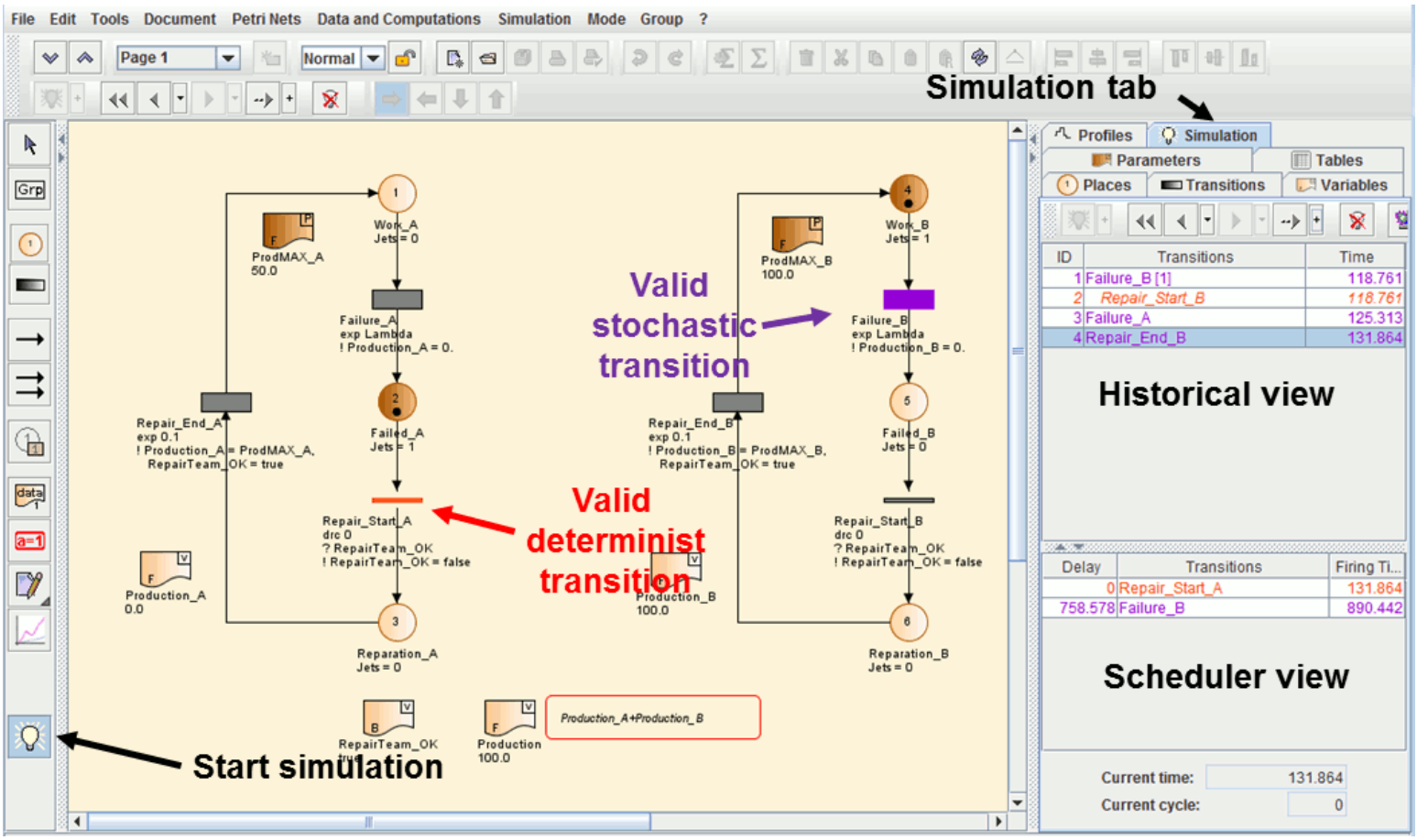
  IEC 61508 specifies "left included" intervals for SIL

10. Simulation interactive

10.1. Introduction

One of the most important characteristics of the GRIF interface is that it enables the user to manually simulate the behaviour of the network s/he has just created. It is therefore easier to understand, debug or explain a model

When the simulation is launched, transitions can be fired to understand how the model reacts, to go back a stage, to replay a sequence of a given event, to view the status of the components, the quantities of flow circulating, the value of each variable, etc. at any time.



10.2. Simulation panel

The interactive simulation panel has four parts:

- Right at the top, a **toolbar** groups together the functions that enable a user to start, stop, configure and play the simulation.
- Just below is a **history** of the fired transitions.
- After that is the **scheduler**, which contains the list of the fireable transitions ranked in order of firing date.
- Right at the bottom, a panel displays the **current time** and the **current cycle**.

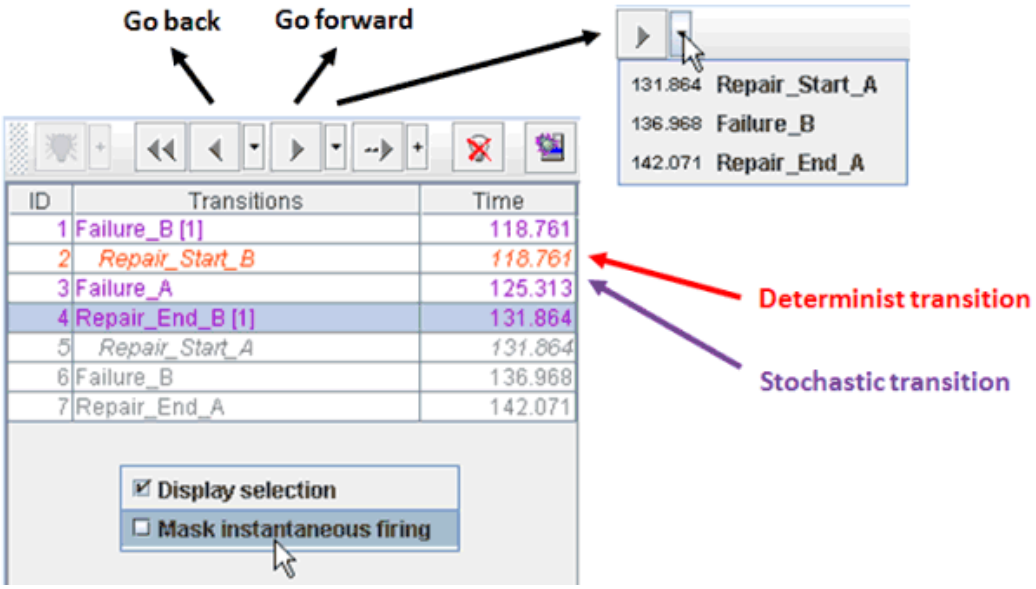
The toolbar contains the following functions:

	Gives access to the interactive simulation mode.
	Opens a simulation configuration window. This point is covered in the section Section 10.5, “Configuring the simulation”.

	Goes back to initial state.
	Goes back one step.
	Rolls back a fired transition. The simulator will go back to the step before the selected transition.
	Replays a step in the fired transition history.
	Replays the fired transition history up to the selected transition.
	Plays the next transition in the scheduler.
	Goes forward in the simulation to a certain time or to the firing time of a given transition. The simulator will stop one step before the transition or time selected.
	Exits simulation mode
	Modifies simulation options during the simulation. See section Section 10.5, “Configuring the simulation”.

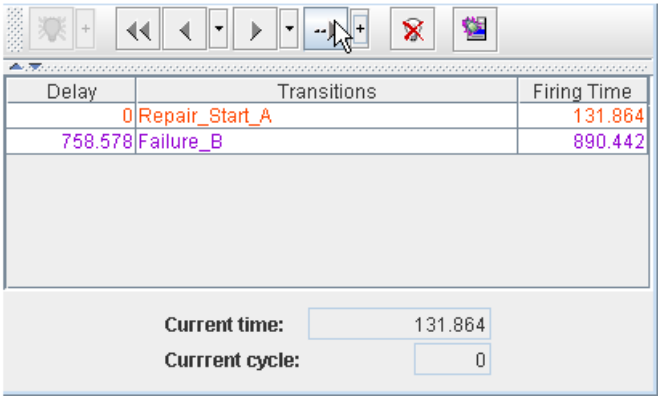
10.3. Simulation history

The simulator contains a panel that displays the history of fired transitions. The user can "explore" the history or take things back a step to modify the running order. A right click on a transition in the history displays a contextual menu that enables the user to display/hide the instantaneous transitions among other operations.



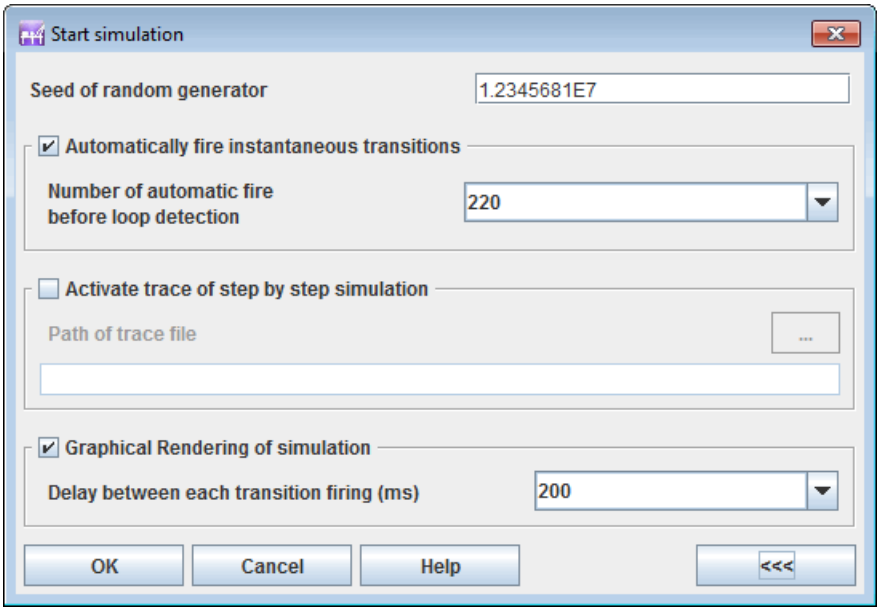
10.4. Simulation scheduler

The simulator includes a panel which displays the scheduler of the transitions to be fired. You can display a transition from the scheduler in the model view by right-clicking on it.



10.5. Configuring the simulation

The simulation options can be accessed using the (+) button located next to the start simulation button, in the simulator toolbar. They can be configured when the simulation is started.



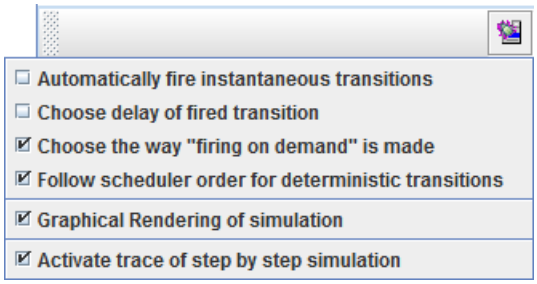
The **seed** to be used by the simulator can be entered here.

The tick box **Automatically fire instantaneous transitions** automatically plays the transitions in Dirac delta function 0. In this case, the user must indicate how many fire operations the simulator can do before considering that there is a loop in the simulation.

The tick box **Activate trace of step-by-step simulation** saves all the transitions fired during the simulation in a file.

The tick box **Graphical rendering of the simulation** enables the user to follow the step-by-step progress of the simulation in the model input area. S/he can then enter the length of time between each transition firing to slow down or speed up the simulation.

Other options can be modified during the simulation



The option **Choose delay of fired transition** enables the user to choose the date on which the transition will be fired (applies only to stochastic transitions).

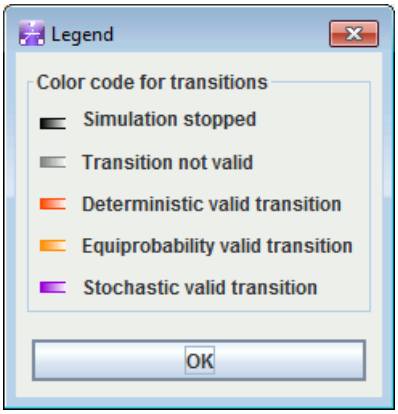
The option **Choose the way "firing on demand" is made** enables the user to automatically fire or not the transitions that use the firing on demand rule. If the option is ticked, a dialogue window will prompt you to select the location that will receive the token if this kind of transition is fired.

If the option **Follow scheduler order for deterministic transitions** is unticked, it is possible to fire deterministic transitions before the date initially planned.

10.6. Colour code / Legend

During a simulation, the transitions active at a given time are displayed in specific colours according to the type of transition.

Below are the colour codes used in the simulator panel for the transitions.

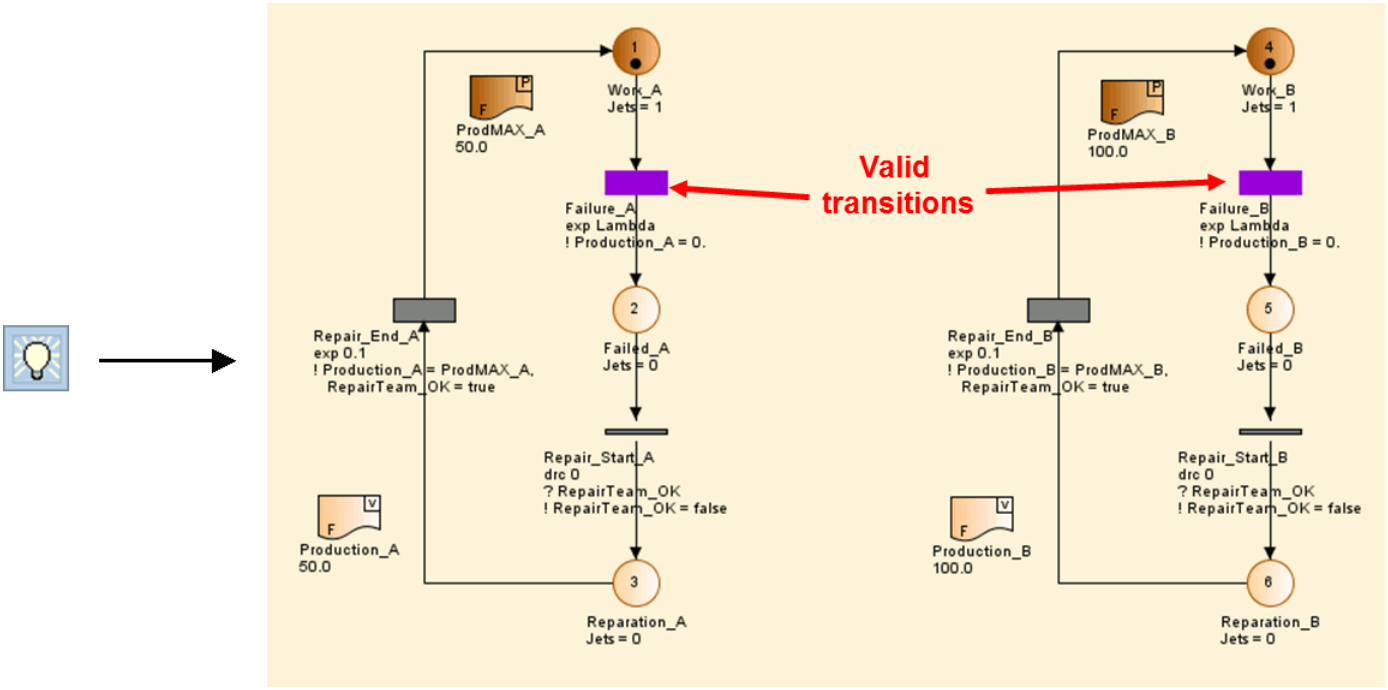


10.7. Transition firing

There are several ways of starting a simulation:

- Use the menu **Mode - Simulation**.
- Click on the corresponding icon in the vertical toolbar.
- Click on the corresponding icon in the simulation panel toolbar.

Once the simulation mode has started, the "valid" transitions in the initial status of the Petri net appear in the scheduler.



In the following picture, two transitions are valid at initial state. There are the failure of right equipment and the failure of left equipment. The two **Failure** transitions are valid because there is one token in their upstream place (**Work**).

This simulation is the expected one: at initial state (system is perfect), the only thing that can happen is the failure of one of the two components.

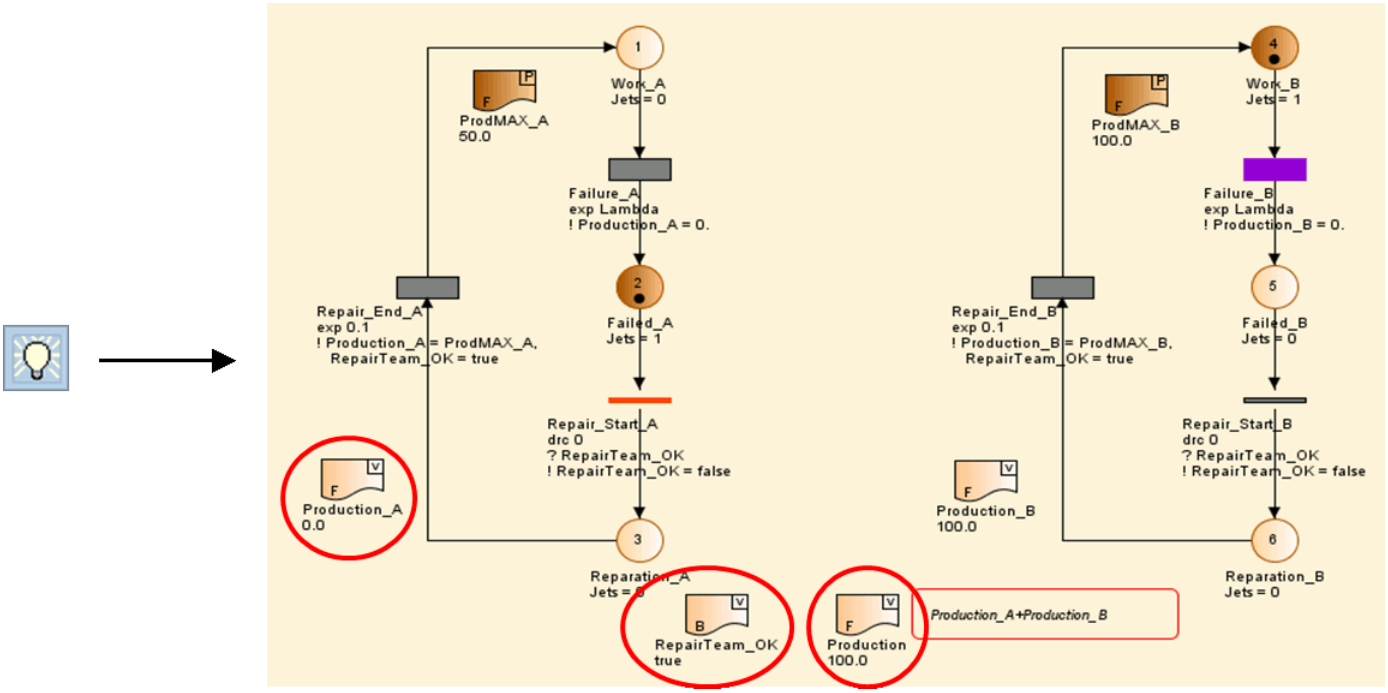
The operation which is of major interest is to be able to "manually fire" the valid transitions:

- choose one of the valid transitions;
- click left on this transition.

The result of this is to:

- remove the token from the **Work** place (which inhibits the **Failure** transition);
- add a token in the **Failed** place.

In the example, given that the Boolean variable **RepairTeam_OK** is initially **TRUE**, the **Repair_Start** transition of the component which failed (that on the left) will be valid. As for the other component, its **Failure** transition will remain valid.

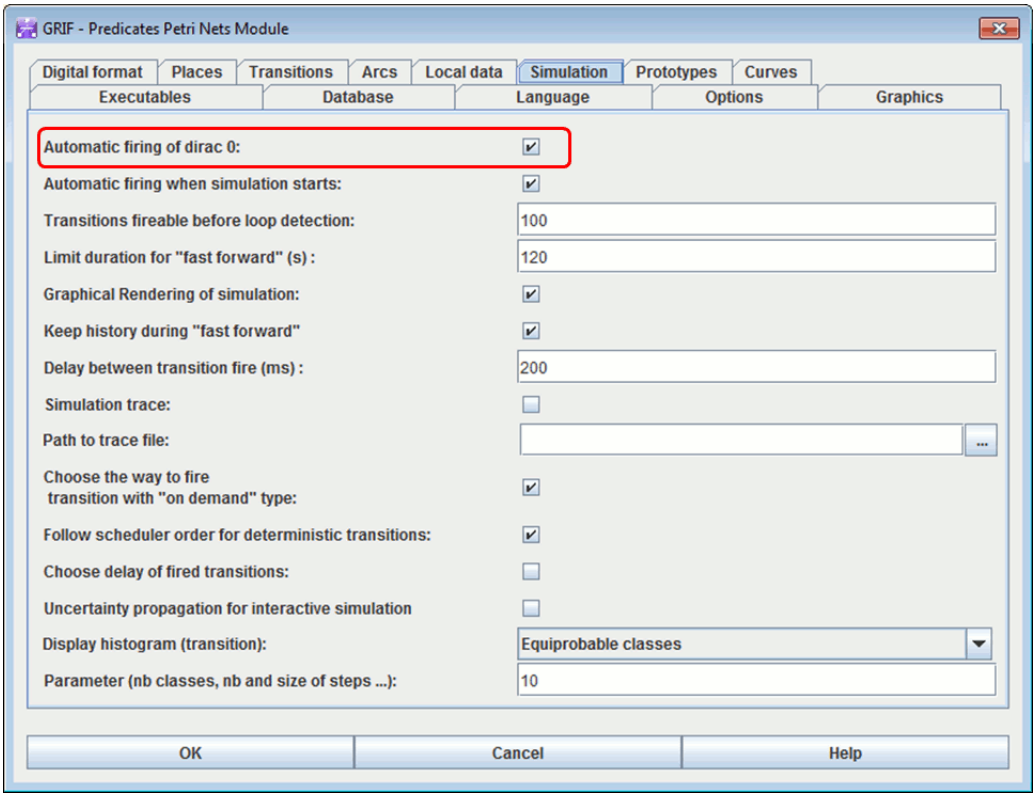


10.8. Automatic firing of transitions with zero delay

The automatic firing of the transitions in simulation mode automatically fires the transitions which have a zero delay (Dirac law with parameter 0) when they are valid. In the case where several Dirac law transitions are in "conflict", the transitions are fired according to their priority, then in chronological order of their creation on a page, then in increasing order of pages. This is how the simulation works when the computations are launched.

Remarque: The transitions with zero delay of a group are fired after those of the page where this same group figures.

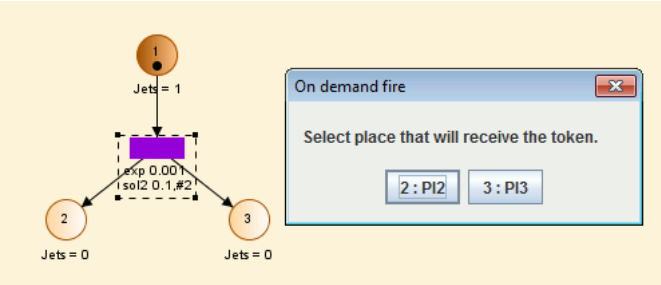
To enable or disable this function go into **Application Option - Simulation**:



10.9. Probability firing for transitions with fire on demand

In simulation mode, when we wish to fire a transition with "fire on demand", the token(s) must only move to a single place downstream. When you click on this type of transition a window is displayed where you have to enter the probability manually (it is normally a probability which is determined by a simulation).

The default value of this probability is 0.5.



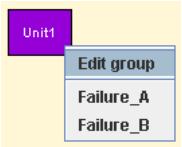
In the above example, if 0.5 is entered, the token will go into place no. 3.

10.10. Simulation with groups

There are two ways of firing one or more valid transitions that belong to a group:

- Enter the group to fire the transitions by clicking on it directly (left click on the group or right click then **Edit group**).

- Or run a fire operation simply using the list of valid transitions in the group and "sub-groups" (right click on the group).



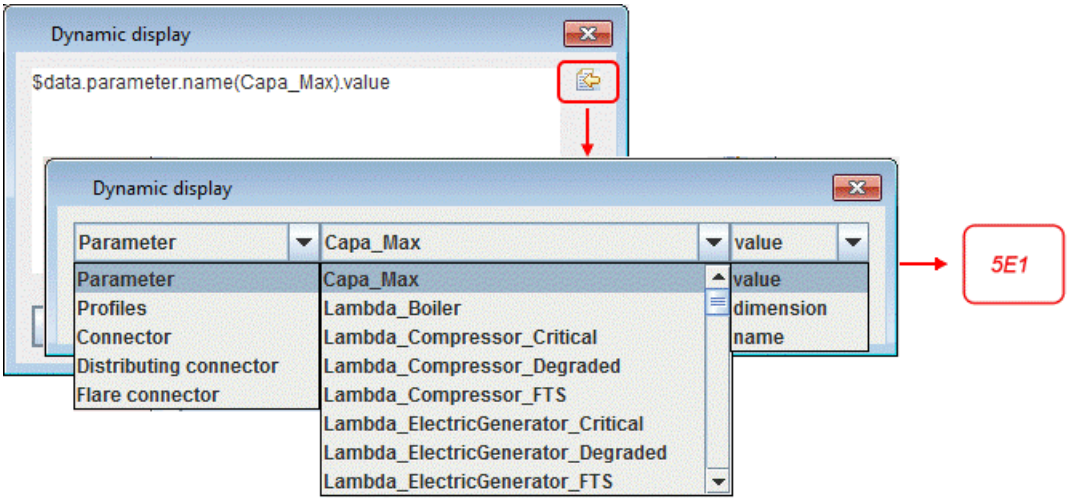
If we go back to the previous example with two components, then the **Failure** transition was effectively the only valid transition.

When transitions are fired within groups, it may prove difficult to determine the modifications that have been made. This is why it is useful to use the dynamic fields.

10.11. Dynamic fields

It may be useful to observe the change in the different parameters of the model. It is also useful to see a result next to its corresponding system. To do this, use dynamic fields by selecting the corresponding icon on the vertical tool bar:

The dynamic fields are a type of "improved comments". They can be used not only to enter words or phrases but also to insert model values or results.



If you want to display information about a data of the model, you must use the following syntax:

$data.'type of data'.'field used o search data'('value that the field must match).'information you want to display for the selected data'

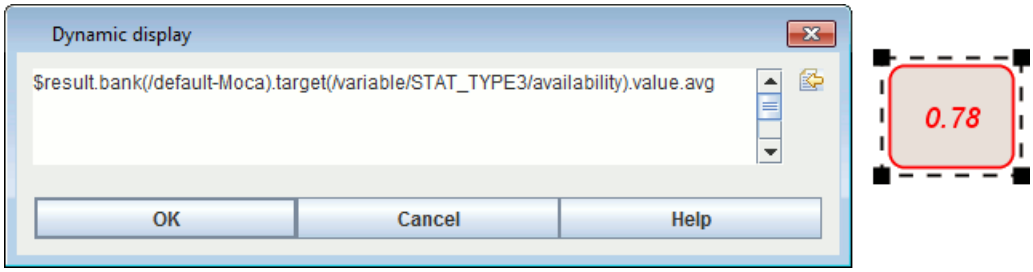
We can analyze the above windows as follows: I am looking for a "parameter" which "name" is 'Capa_Max", and I want to display its "value". When you type the first letters, a completion system helps to type script without error.

Button in right permits to enter complete expression but select what you want to appear.

If you want to display a result of the result-bank, the syntax is the following:

$result.bank('path in the bank').target('target result').'what you want to display'.'at what time'

We can analyze the next picture as follows: I am looking for a result which path in the bank is "default-Moca", I want results for "TS3 for 'available' valriable" and I want its value for the "last" time. If last is replaced by time(10) we obtain value at t=10.



You can also display a summary of result. Replace 'what you want to display' by **summary**. In this case, **summary** is the last word of this script.

11. Computations

11.1. MOCA computations

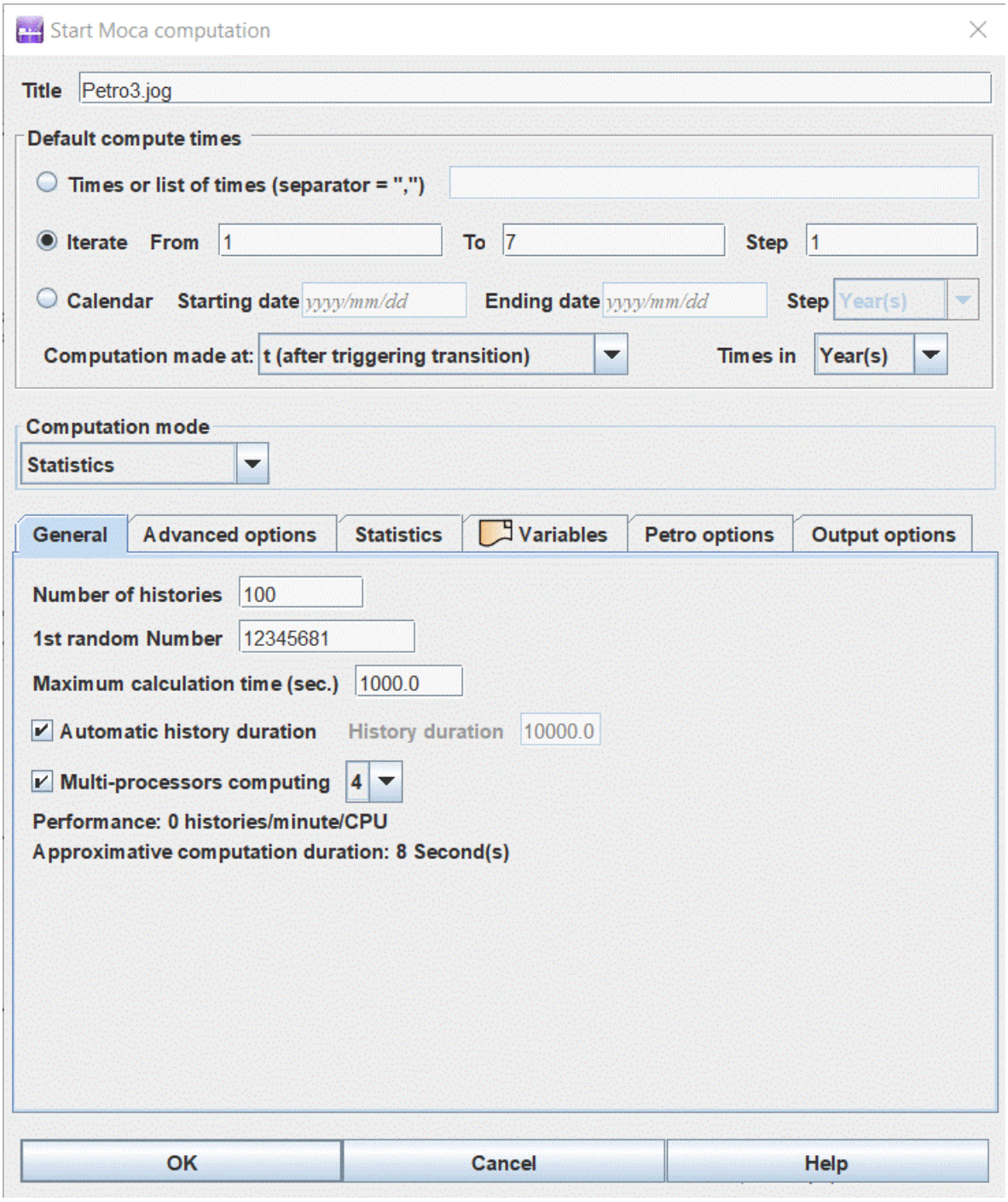
The computations using MOCA-RP V14 are performed in three main steps:

- general configuration of parameters;
- the launch itself;
- reading the results file.

11.1.1. Configuring the computations

The computation configuration window can be accessed in two different ways: either via menu **Data and Computations - Moca Data** or via **Data and Computations - Launch Moca ...** . The difference between the two is that, in the second case, the configuration step is directly followed by the computation launch step.

The configuration window which opens is called **General Information** :



This configuration window is divided into five parts:

1. **Title** : enables you to give a title to the results file.
2. **Default computation times for statistic states** :
   - **Iterate From** A **to** B **step** C: the computations will be performed for values of t ranging from A to B with a step of C.
   - **List of times** : the computations will be performed for the values of t given in this list.
   - **Computation made at** : by default, computations are made immediately after transition triggering, but you can do computation at t-Epsilon (just before triggering), or at both.
   - **Unit** : default times unit is "hour". You can choose a unit that will be used for computation times. N.b. results are always in hours.

3. **General** :

- **Number of histories** : Number of histories (NH) to be simulated (each history has a time t indicated below).
- **First random number** : It is the seed of random number generator.
- **Maximum computation time** (MT): The computations are stopped and the results are printed even if the requested number of histories has not been reached.

the unit of time (MT) is the second.

- **Automatic history duration** : If this box is checked, GRIF will compute history duration using computation time of variables and statistical states. If not, user can choose a specific **History duration**
- **Multi-processors computing** Enables (or not) the multi-processor computing (when available).

4. **Advanced options** : used to configure the advanced options.

- **Loop detected when the number of transitions firing at the same time is grater than** : You can choose the limit of transitions fired at the same time before loop detection.
- **Max number of loops while()** : if using in a transition.
- **Continue calculation if errors** : If an error is detected, the current history is stopped and we pass to the following one without stopping the calculations.
- **Display seed history** : Display or not the seed used by the random number generator.
- **Use old seed management (version < 2018)** : From 2019, a new random number generator was developed to increase the number of playable story without seed repetition. It is recommended that you use this option only for backward compatibility for your old documents.
- **Computes every nodes** : Useful option in the boolean modules to have the calculations on all the nodes (by default the calculations are performed only in the top event).
- **Delay re-computation for dynamic transitions** : Selects the method for recalculating the delay of transitions marked as dynamic. For more information on the possible choices, refer to User manual Moca14.10.3 [2.7.10. Dynamic transitions]

5. **Statistics**
   - **Confidence interval** : Allow to choose the confidence interval at 90%, 95%, 99%, 99.9% and 99.99% (by default 90% is checked). .
   - **Disable statistics on places and transitions** : Allow not to display the statistics on places and transitions.
   - **unlimited histogram** : In case of histogram all the stories are taken into account not only the 1000 first ones.
   - **Do not save 0 value in history** : Option to not save the historic if the statistic value is 0.
   - **Activate uncertainty propagation** Enables (or not) the uncertainty propagation computations (two-stage simulation): in this case we must specify the number of sets of parameters "played" (the real number of histories thus simulated will be the "number of sets of parameters x number of histories to be simulated" and will be displayed in the "Total number of histories" field).
   - **Histogram with every history of the tries (with uncertainty propagation)** : Allows to display the histogram with every history of the tries.

6. **Variables** : This tabs reminds computing configuration of variables. If document contains some statistical states, another tab is available.

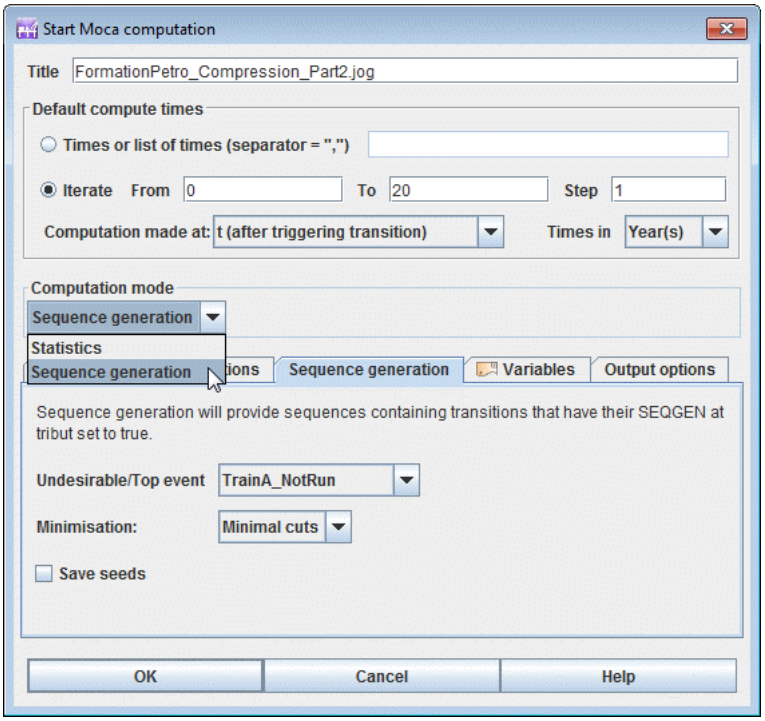
7. **Output options** : used to configure the output.

- **Print censured delays** : Prints the censored delays (or not).
- **Verbose** : Give internal information of Moca RP (launching, compilation, ...).

11.1.1.1. Sequence generation computations

The MocaRP computation engine enables to perform sequence generation. In this computation mode, it is possible to list all the sequences of transition fire which leads to undesirable event. However, statistical computation will not be made.

To activate this computation mode, in the option in the computation launching, select **Sequence Generation** in **Computation mode**. **Statistics** tab disappears in favour of **Sequence Generation** tab.



In the **Sequence Generation** tab, it is necessary to indicate the name of the **Undesirable event**. This event is a variable defined by a boolean expression. A `true` value indicates that the undesirable event has occurred during the current story.

Sequences that result in the undesirable event are grouped into an equivalence class. It is possible to choose the algorithm used to define this equivalence class:

- **None** : All transition sequences leading to the undesirable event will be returned. (no minimization)
- **Sub-words** : The minimality consists to search a common prefix, factor or suffix.
- **Minimal cut sets** : only the transitions are interested, without duplicates and without order, which led to the undesirable event. A minimal cut set includes all the sequences that include it in the same equivalence class.

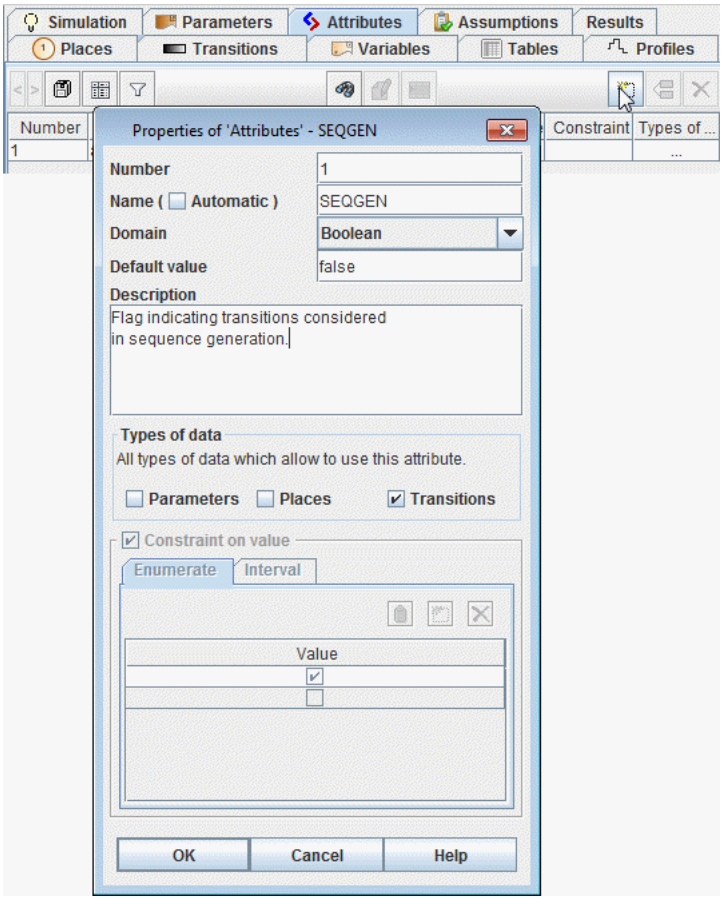
See Manuel utilisateur Moca14.10.3 [3.4.5. Lancement d'une génération de séquences] for more information.

The computations options **Save seeds** enables to save the history seed and the occurrence time when undesirable event occurs.

Only the transitions with the flag `SEQGEN` with a true value are considered in the sequence. All others transitions are not taken into account. They generally correspond to transitions that don't contribute to the sequence generation (such as instantaneous reconfiguration transitions).

By default the transitions are not marked as being able to be part of the sequences. To take into account a transition in the sequence generation:

- Create a boolean attribute called `SEQGEN`.



- Define `true` value of the attribute `SEQGEN` for transitions considered in the sequence.

11.1.2. Reading the results

The results are presented in a window with different tabs and tables.

11.1.2.1. Moca Results

Moca results atre displayed in a window containing 6 main tabs: variables, places, transitions, XML, stantard output, info.

11.1.2.1.1. Tab of Variables

The **Variables** tab contains all the information computed for each variable (or statistical state).

- **Value**: Contains every value of a variable for every type of statistic.
- **History (at the end of histories)**: contains historical values for each computed statistic.
- **Fixed size Histogram**: Contains histograms computed by Moca (cf. chapter about histograms for Simulation package module)
- **Equiprobable classes Histogram**: Contains histograms computed by Moca (cf. chapter about histograms for Simulation package module)
- **User defined Histogram**: Contains histograms computed by Moca (cf. chapter about histograms for Simulation package module)
- **Timeline**: Contains a timeline for each variable. Times are automatically computed by Moca.

11.1.2.1.2. Tab of Places

It contains sojourn duration and mean mark for each place of Petri Net.

11.1.2.1.3. Tab of Transitions

It contains firing frequencies for each transition, and firing history for each history.

11.1.2.1.4. Other tabs

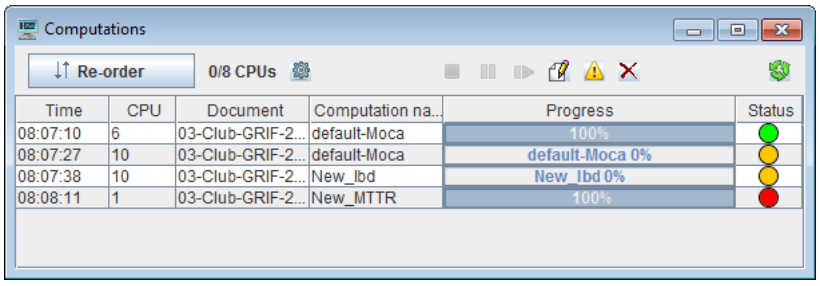
Other tabs display "raw" results. XML tab contains XML output of Moca, it is the file used to retrieve data. This file can be used for further post-treatments.

Standard output displays the standard output of Moca (available only after computing).

Info tab contains useful information about computation (simulation time, number of histories that have been done ...). This information can be copied / pasted for any other use.

11.2. Compute manager

Compute manager shows the calculations. That are currently running or already performed.

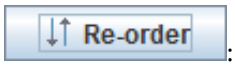


Compute manager is automatically displayed when calculations are performed. User can display the window using the following icon .

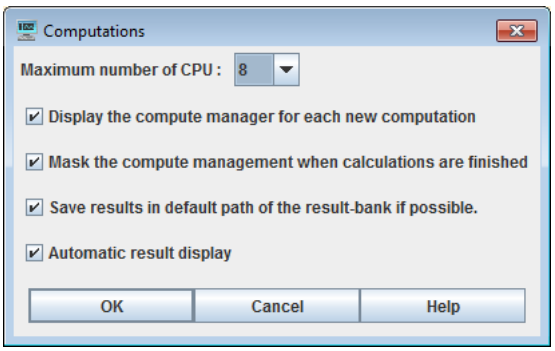
This tab is made of 6 columns:

- **Time**: The hour of calculation launch;
- **CPU**: number of CPU used;
- **Document**: document name;
- **Computation name**: name of results file;
- **Progress**: progress bar;
- **Status**: finished in green, in progress in yellow, error in red;

In **Compute Manager** some actions are available:

- 

: allow to reorganize the calculations order;
- : display the following windows for computation settings:

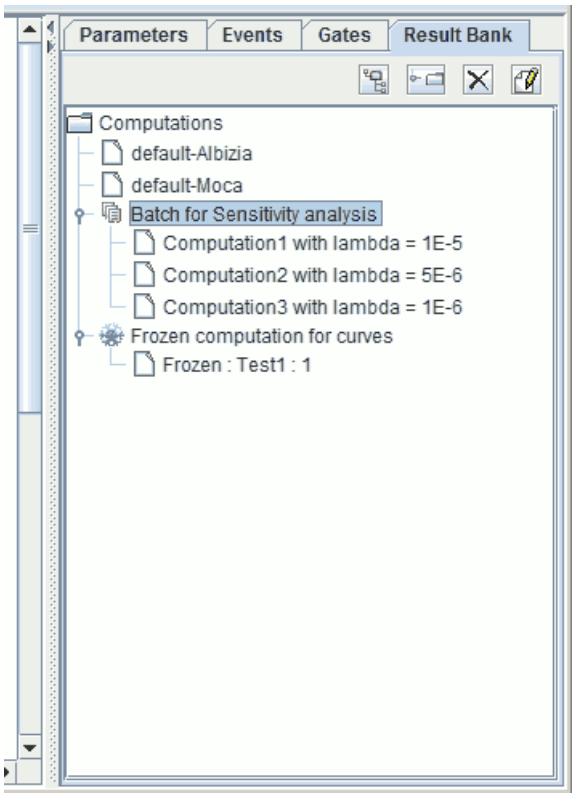


- : stop selected compute;
- : suspend selected compute;
- : resume compute in suspend;
- : display results of selected compute;
- : details errors;
- : remove selected compute;
- : clear all compute;

When a task is added to Compute manager, user is not blocked until the task is ended. He can continue to work on his model. He can even relaunch a calculation. The various tasks accumulate and are treated sequentially.

11.3. Result Bank

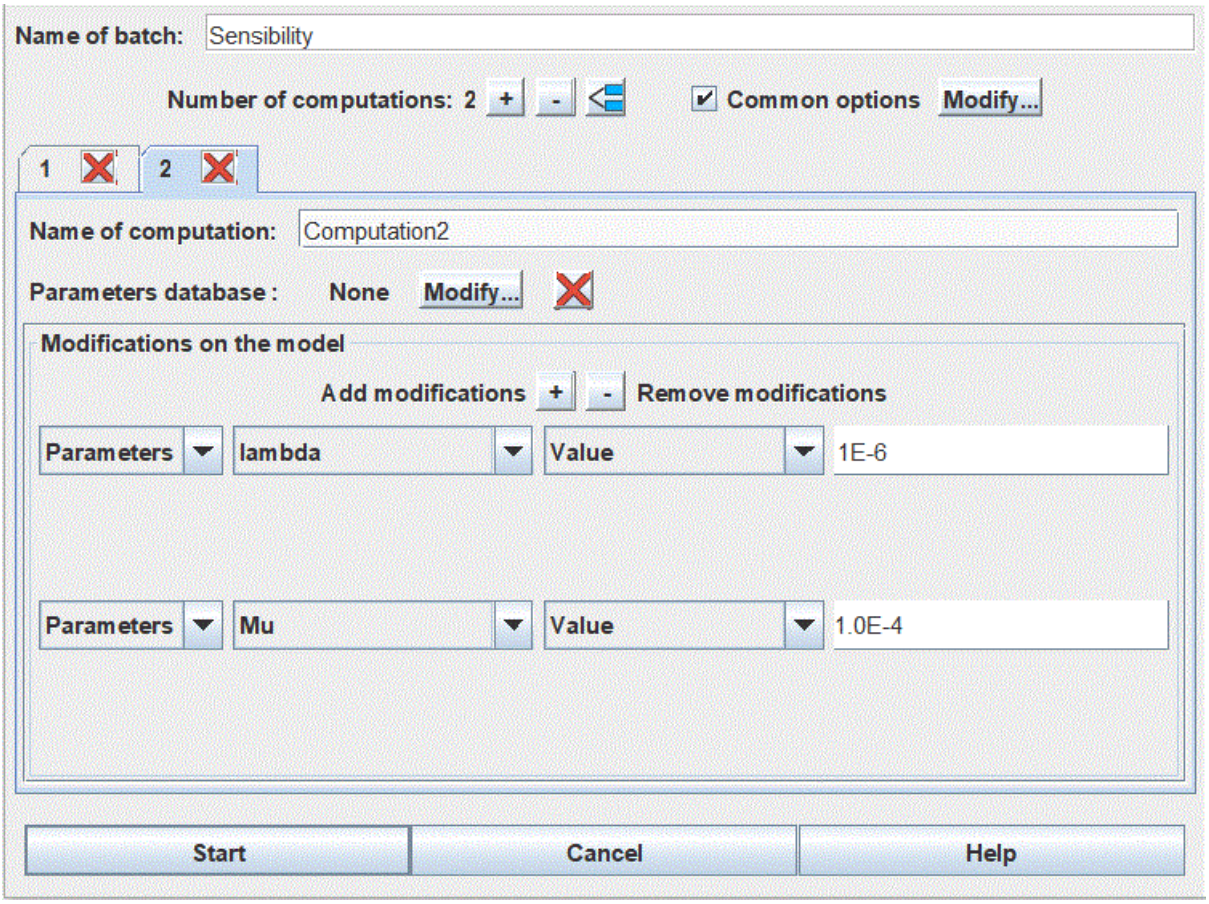
Every GRIF computation is stored in result bank which is available on the right of the module.



You can display a result with a double-click on it. There is a default result for each computation engine, it is the place where "standard" computations are stored. Then, each is a batch computation directory, it contains as many results as computations asked for the batch. Finally, the directory contains results for curves that have been frozen.

11.4. Batch computation

In order to do fast sensibility analysis or to compare some results with different parameters, it can be very useful to do calculation one after another automatically. To do this, use the **Data and Computation / Batch computation ...** menu.



The batch launching window is made of two part, the first is for the name of the batch and the number of computations in the batch. Then each computation can be set up:

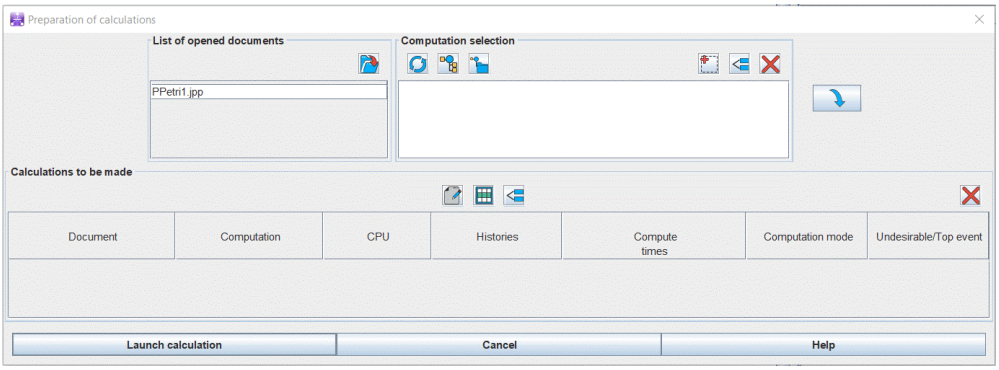
- **Name of computation:** for identification in results
- **Computation options:** contains every option related to this computation (times, types ...)
- **Modifications on the model:** specifies modifications that will be made on the model before computation launching. You can add as many modifications as you want with the + button. Each modification is made of 4 parts:
  1. 1 drop-down menu for object type
  2. 1 drop-down menu for the object that will be modified
  3. 1 drop-down menu to specify what will be modified on this object (value for a parameter, law for other object ...)
  4. Then you must enter the new value in the cell

The above example shows a batch with 2 computations, the first is made with a lambda parameter with 1.0E-5 value and a mu parameter with 0.1 value.

After a computation, the model is always reset up to an initial state without modification.

11.5. Preparation of calculations

by clicking on the icon user has access to **Preparation of calculations** .



This window is made up with 3 parts:

- the left top part indicates the **list of opened documents** on which the calculations will be performed. It is possible to load a new document using the icon ;
- The right top part enables (once the document is chosen) to select the bank of result where the calculations will be record. The following options can be used:

  – enables to add a bank of results;
  – enables to duplicate a bank of results;
  – enables to delete a bank of results;
  – enables to expand the different banks of results;
  – enables to collapse the different banks of results;

  It is then enough to press on the button to configure the calculations to be made.
- The bottom part enables to specify the options of wanted calculations.

  – edits a computation;
  – gives the access to the multiple changes;
  – duplicates a computation;
  – deletes a computation;

12. Curves

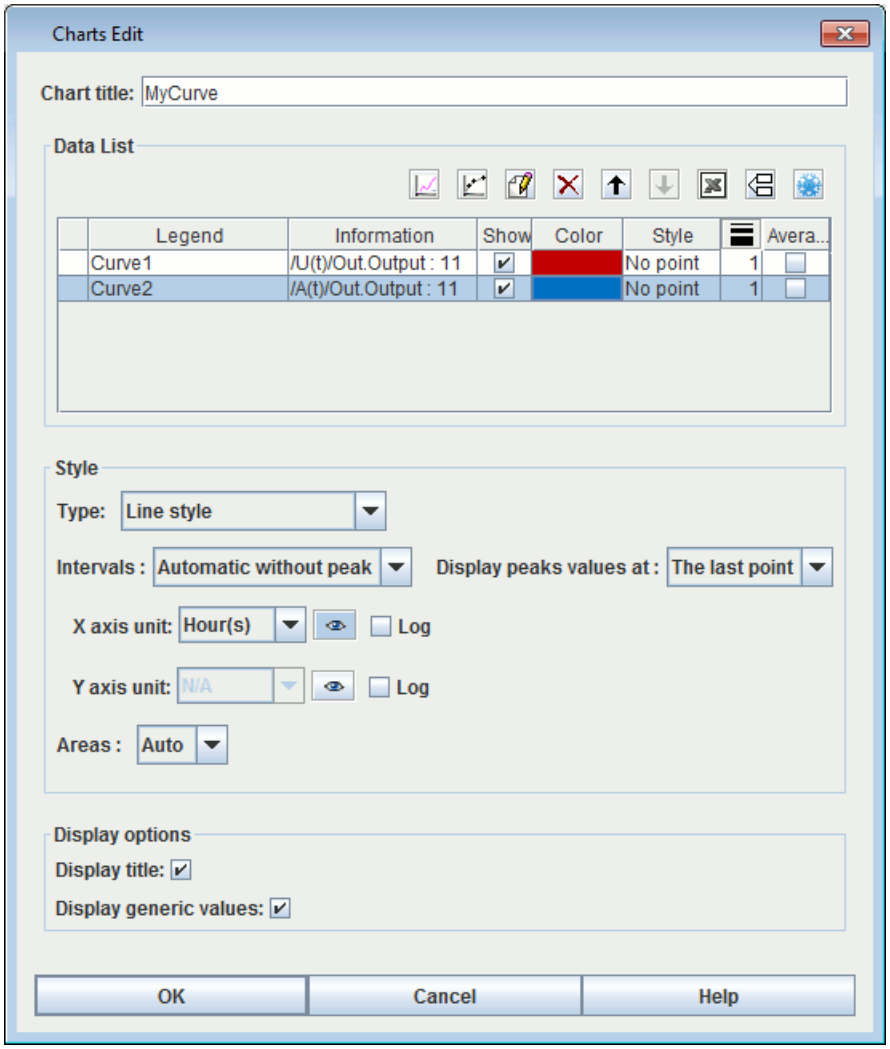
The curves can be drawn to study the model and the results better. To do this, click left on the corresponding icon on the vertical task bar then draw a box. This box will be the space assigned to displaying the curve(s). Initially it is only a white box with two axes without graduation.

Charts icon:

We must now define the curves to be drawn. To do this, click right on the box to display the **Charts Edit** window.

12.1. Edit curves window

The edit curves window is the same for all the GRIF modules.



The window is divided into several sections:

1. **Charts title**: enters a title for the graph.

2. **Data list**: this part includes a table with several columns in which the different curves on the graph are listed (name, description, display, curve colour, curve style, curve thickness, display average). A number of different buttons are available above this table.

- : Selects a result of computations to display. It sends the user back to the Select results window to add a curve plot to the graph (see. Section 12.2.1, “Curves from data in result-bank”).
- : Compares several results from different calculations for the same data. It sends the user to the Compare results window to add a curve plot to the graph (see Section 12.2.2, “Comparative curves from data in results bank ”).
- **Edit**: edits the plot of the selected curve.
- **Delete** : deletes the plot of the curve selected on the graph.
- **Up** : moves up the plot of the curve selected in the list.
- **Down** : moves down the plot of the curve selected in the list.
- **Save**: saves the list of points calculated to plot the selected curves in .csv format. This export does not contain the generic values. To obtain an export with the generic values, right click on the curve and select **Individual export**.
- **Duplicate**: creates a new curve identical to the curve selected.
- **Freeze** : freezes the display of the curve, which will no longer be updated automatically according computation results.

For each curve, the user can specify the colour, point style, line thickness and display.

3. **Computation options**: enables the user to enter settings for the computation (optional depending on the module).

4. **Style**: this section concerns the curve display.
   - **Type of style**: specifies the type of all the curves on the graph (line, histogram, etc.). N.B. In the case of a histogram, the bars that exceed the display area will be displayed in shading to show the user that s/he must change the display intervals so that the entire bar can be displayed.
   - **Intervals** defines the display limits for the curve. **Automatic without peak**: the graph will not display the "peaks" of specific cases of exceptional values which would make the graph illegible. Even if the peaks are not displayed on the graph, the user can display their values using the choices proposed in the option **Display peak values**.
   - **X and Y intervals**: specification of the display interval on the X and Y axes (default intervals or intervals defined by the user). The last function enables users to "zoom in" on the most interesting parts of the graph.
   - The axis unit can be selected according to the type of computation result. For example for units of time, the user can enter hours, days, months or years.
   - The button determines whether or not the unit is displayed on the graph.
   - The **log** boxes are used to activate the logarithmic scale on the axis in question. N.B. 0 cannot be represented on a log scale, remember to enter a start value that is strictly positive (e.g. E-10). If 0 is entered, the log scale will begin at an arbitrary value E-15. Where the computation engine enables, the **trust interval** can be displayed by ticking the corresponding box.
   - **Areas**: distinguishes a range of values on a coloured background.
   - In the histogram style, a box can be ticked to create a cumulative histogram.

5. **Display options**: activates the **Display title** function (display graph title) and the **Display generic values** function (display min, max and average for each curve).

When a curve is edited, the edit curve window often includes 3 parts: the times at which the computations are performed, what has been calculated and the extra information (generic values) that should or should not be displayed below the curve.

Comment: it is sometimes necessary to refresh all the graphs in a document. This can be done using the **Tools / Refresh** command, or the keyboard shortcut F5 or the icon.

12.2. Selection of results window

12.2.1. Curves from data in result-bank

When you click the **Add** button in the **Data list** part you reach a window for curves setup. Each curve displays data stored in the result-bank. The following window helps users to specify how to retrieve data.

- **Legend**: legend of the curve.
- **Computation selection**: select the computation in the result-bank.
- **Result to be displayed**: each computation contains many results. Select the one you want to be drawn.
- You can display a list a point whose X and Y will be taken from the data selected in the combo-boxes; or you can display the time spent in each zone.
- **Axes**: When a result is selected, select what must be in X-Axe and what must be in Y-Axe.
- **Value to be displayed**: Then you can display additional information about the result (min, max, average ....)

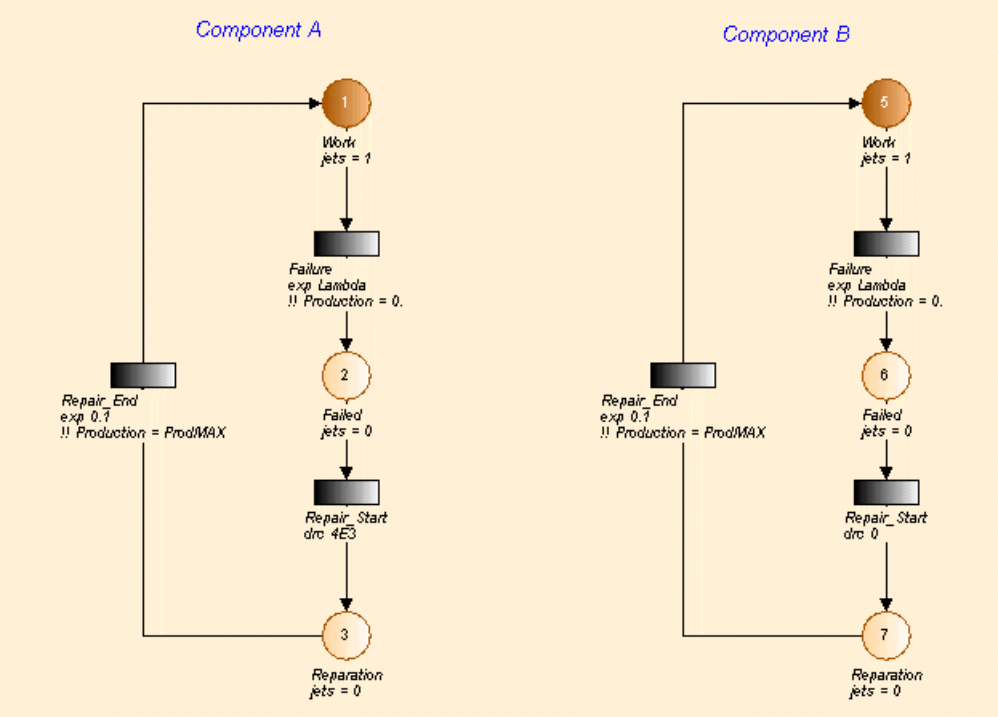
12.2.2. Comparative curves from data in results bank

When the user clicks on **Compare** in the **Data list** section, a window opens enabling the user to specify the curve to be plotted. S/he can then choose whether or not to display a result from several different computations. The following window is used to indicate the way in which the information is to be recovered.

- **Legend**: curve legend.
- **Computation selection**: the user can select the different computations to be used from the results bank. Hold down the Ctrl button to select several different computations.
- **Result to be displayed**: a computation often contains several results and this file tree structure is used to specify the result that the user wants to display.
- **Information to show**: indicates which data are to be displayed on the y-axis.
- **For which point?**: indicates which point of the computation is to be compared to the others.
- **Display x-axis values according to**: indicates which data must be displayed on the x-axis.
- **Value to be displayed**: finally, certain extra data can be displayed (min, max, average, etc.)

12.3. Examples of curves

Here are the Petri Nets from which the different curves will be drawn.

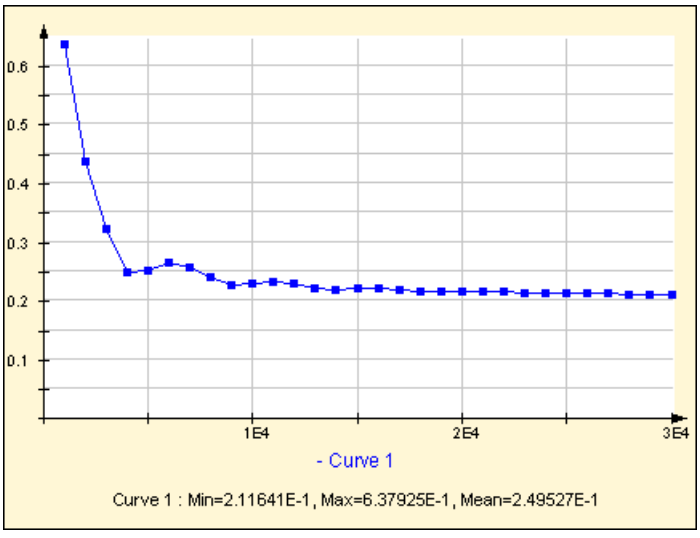


There are two components, A and B:

- A can be repaired with a delay before repair modelled by a Dirac law transition.
- B can also be repaired but with no delay before repair.

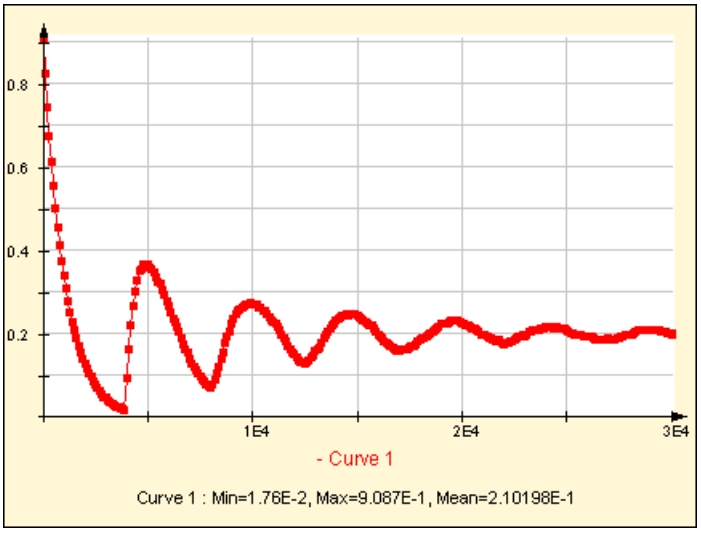
The simulation will be carried out for 10,000 histories of 30,000 hours.

12.3.1. Availability



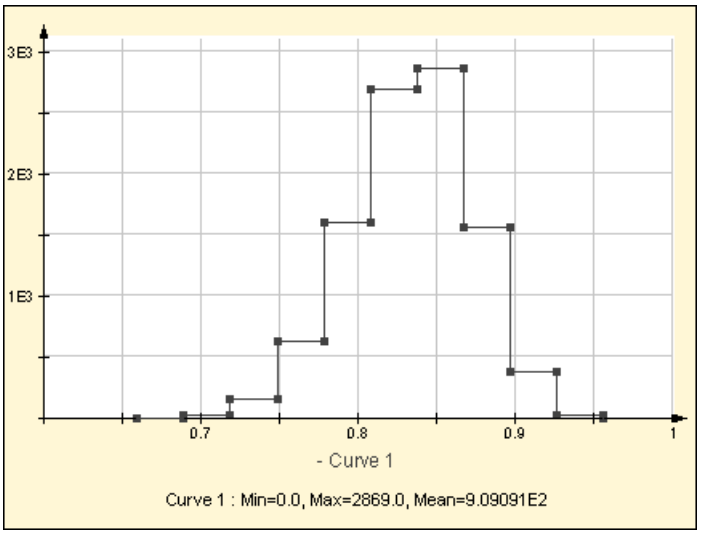
This curve represents the change in mean availability of component A over time. The points have been computed at regular intervals (1 every 1000 hours).

12.3.2. Timing Chart



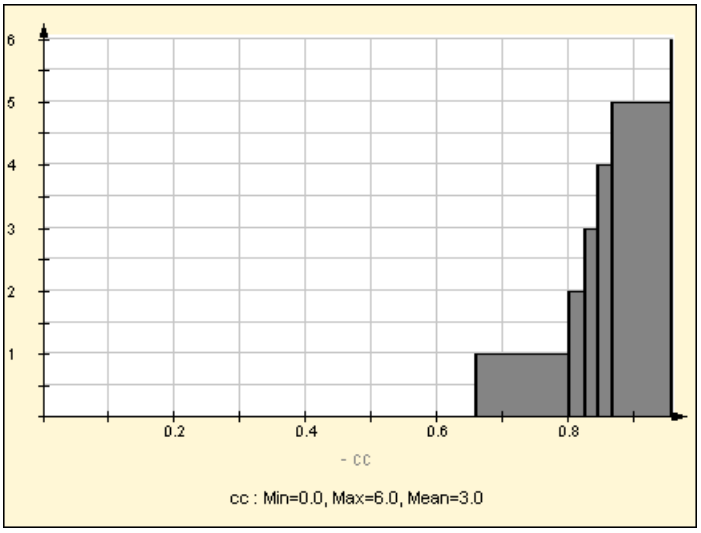
This curve also represents the change in instantaneous availability of component A over time (marking of place no. 1). In this case the points have been computed according to the curve variation. This "captures" the discontinuities better and thus makes the curve "smoother" and more precise.

12.3.3. Fixed Size Histogram



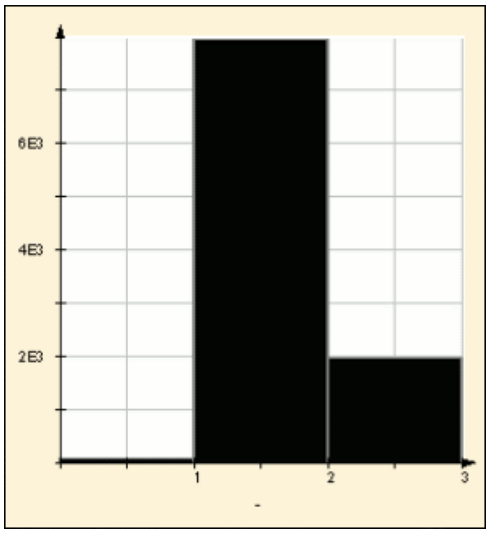
This histogram was produced from the mean value of the availability of component B (marking of place no. 5). The 10,000 results from the 10,000 simulated histories have been stored in 10 classes of the same interval.

12.3.4. Equiprobable Classes Histogram



This histogram was produced from the mean value of the availability of component B (marking of place no. 5). Five equiprobable classes were requested. The probabilities that component B's mean availability is in one of these classes is identical.

12.3.5. Defined Interval Histogram



The 10,000 results from the 10,000 simulated histories have been stored in defined intervals. We have defined "[0,1,2,3" (equivalent to [0,1[[1,2[[2,3[cf. Moca User Manual for more information) We have computed number of working components at the end of histories. We see that 2 components works at the end of histories most of the times. Intervals corresponding to S.I.L can be defined in order to see in which S.I.L component is for different stories.

Bounds of intervals can be defined as follows:

- Automatically defined limits for SIL
- Manual definition of limits (separate by commas)
- "Iteration": user choose lower limit and upper limit, and the size of intervals.
- "Iteration (log scale)": user choose lower limit and upper limit, and the number of intervals. Size of intervals will be computed in order to have same-size-intervals, on a logarithmic scale.

Moreover, two limits are added at minus infinity, and plus infinity, in order to have a chart containing every history of the simulation.

When limits are chosen, user has to choose between "left included" or "right included".

IEC 61508 specifies "left included" intervals for SIL

13. Database of parameters

In every GRIF module, a connection can be established with a database of parameters, to import parameters in GRIF. There are three ways to connect to a different database:

- connection to a .csv file
- connection to a .xls file
- other connection (via JDBC).

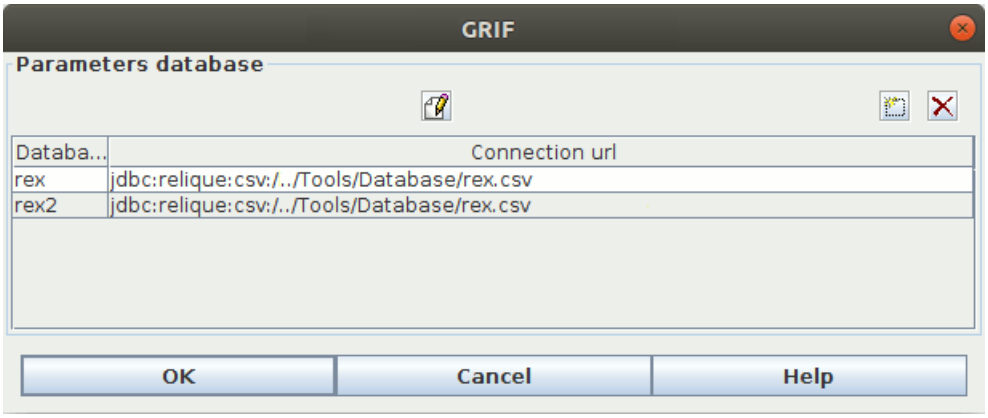
13.1. Format of the databases

The database must contain the identifier, the name and the value of the parameter. It is possible to add to the parameters more information, as the unit, the dimension and the description of the parameter. So we can have three to six columns, inquiring:

Data's type:	Possible values:
Parameter's identifier	Numbre, Text
Parameter's name	Text
Parameter's value	Number
Parameter's description	Text
Parameter's unit	HOUR : hours
	DAY : days
	MONTH : months
	YEAR : years
	HOUR_1 : $hours^{-1}$
	DAY_1 : $days^{-1}$
	MONTH_1 : $month^{-1}$
	YEAR_1 : $years^{-1}$
	FIT : Failure In Time (= 10^{-9} $hours^{-1}$)
Parameter's dimension	BOOLEAN, FACTOR, PROBABILITY, RATE, TIME, OTHER

13.2. Connect to a database

To access to the window to create the connections to databases, go to the menu **Data and Computations -> Parameters database -> Connections ...**. A window appears then:



From this window, it is possible to :

	Add a connection to a database.
	Modify a connection to an existing database. It opens the same window when adding a connection, but the fields are already filled by the data previously entered.
	Delete the selected connections of the databases.

13.2.1. Connection to a CSV file

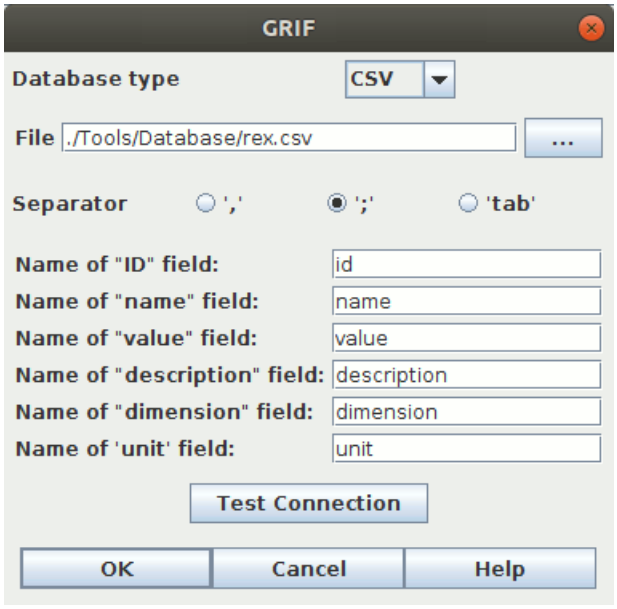
13.2.1.1. Form of the database

This type of connection is the simplest. The CSV file has for extension ".csv". It is a simple text file where the different fields are separated by commas, tabulations or semicolons.

```
ID,NOM,VALEUR,DESCRIPTION,DIMENSION
000001,Lambda,0.001,Exemple de Lamda,RATE
000002,Mu,0.01,Exemple de Mu,RATE
000003,Gamma,0.5,Exemple de Gamma,PROBABILITY
000004,ProdMax,1000.0,Exemple de Production maximum,OTHER
```

13.2.1.2. Connection

Once clicked on the button "Add a connection to a database", a window opens up:



This window has as a common base, the selection of the database, the fields for "ID", "name", "value", "description", "dimension" and "unit", and a button **Test Connection**. By clicking on this button, GRIF tries to connect to the database and so verifies the configuration provided by the user.

When adding a CSV database, the type **CSV** must be selected. A new field appears: the separators between the data. To sum up, there are three steps to add a connection to a CSV database:

- First, fill the path of the CSV file in. A file explorer is at your disposal (button **...**).
- Then, specify the type of the separators used in the CSV file.
- Finally, enter the six fields names of the CSV file. (Or only the ID, name and value fields) (Uppercase letters are taken into account as lowercase)

Warning : It's important to note that when creating a connection to a CSV database, you must have all of the data on a single sheet.

13.2.2. Connection to a XLS file

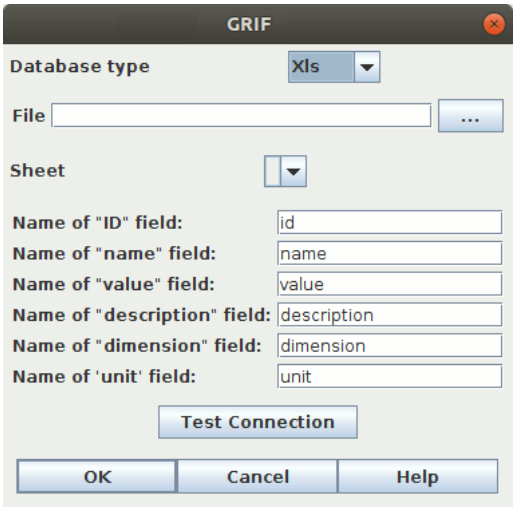
13.2.2.1. Form of the database

The databases of the .xls or .xlsx extensions correspond to EXCEL files. Here is an example of an EXCEL Database :

	A	B	C	D	E	F
1	ID	NOM	VALEUR	DESCRIPTION	DIMENSION	
2	1	Lambda	0.001	Exemple de Lamda	RATE	
3	2	Mu	0.01	Exemple de Mu	RATE	
4	3	Gamma	0.5	Exemple de Gamma	PROBABILITY	
5	4	ProdMax	1000.0	Exemple de Production maximum	OTHER	
6						
7						

13.2.2.2. Connection

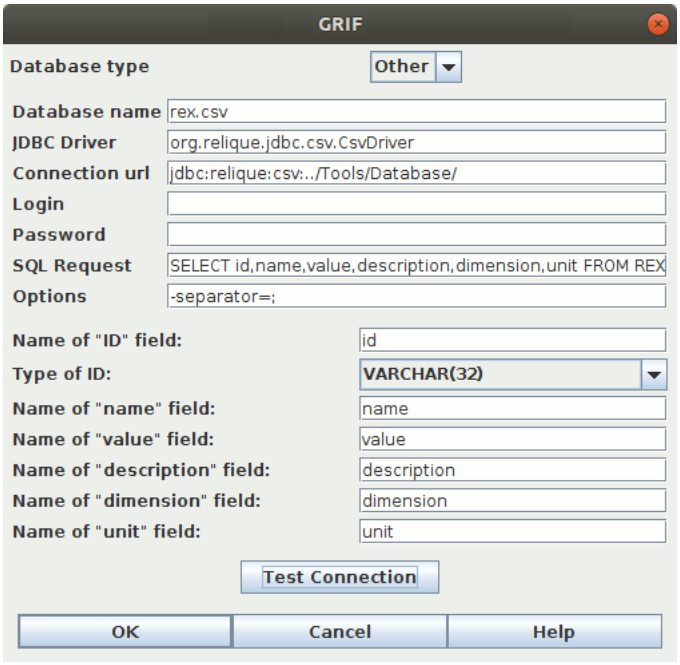
To connect GRIF to this database, select the **XLS** type in the connection window. The window is now as followed:



Sheet is the sheet's name where the data are located, and will be filled once a valid path to an EXCEL file has been entered.

13.2.3. Connection to a database (with a JDBC connection)

GRIF can connect to any database with JDBC, as long as the database follows the same rules of the databases seen earlier. The window for that kind of connection has multiples fields to fill:



1. **Driver JDBC** is the name of the JDBC driver (ex : sun.jdbc.odbc.JdbcOdbcDriver)

2. **Connection URL** is the URL of the database.

3. The fields **Login** and **Password** can be left empty.

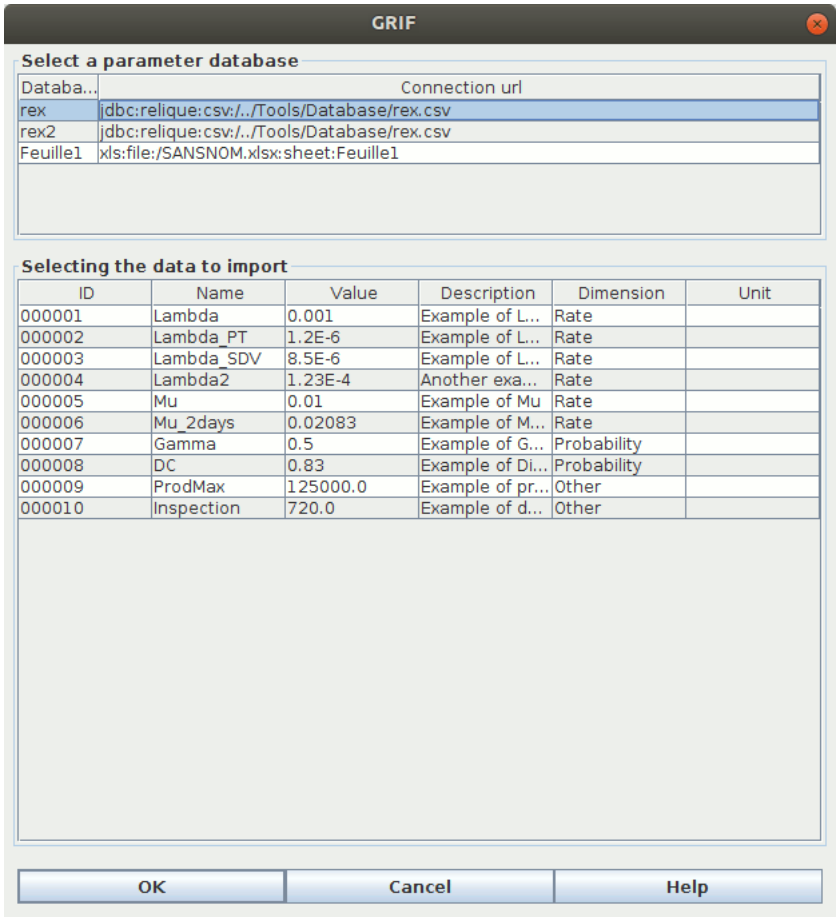
4. The SQL request **SELECT id,name,value,description,dimension,unit FROM REX** is used to gather the dates.

5. **Option** field inform of all of the database's options: separator, ...

Once a connection with a database is ready, GRIF can now import a set of parameters from the database, but also updates these parameters when modifications has been made in the databases, or recreate the links of these parameters so they can now take the values of another database.

13.3. Import parameters from a connected database

Once a database is connected, GRIF can import a set of parameters from the database, via the window reachable by the **Data and computations -> Parameters database -> Copy parameters from database ...** menu.

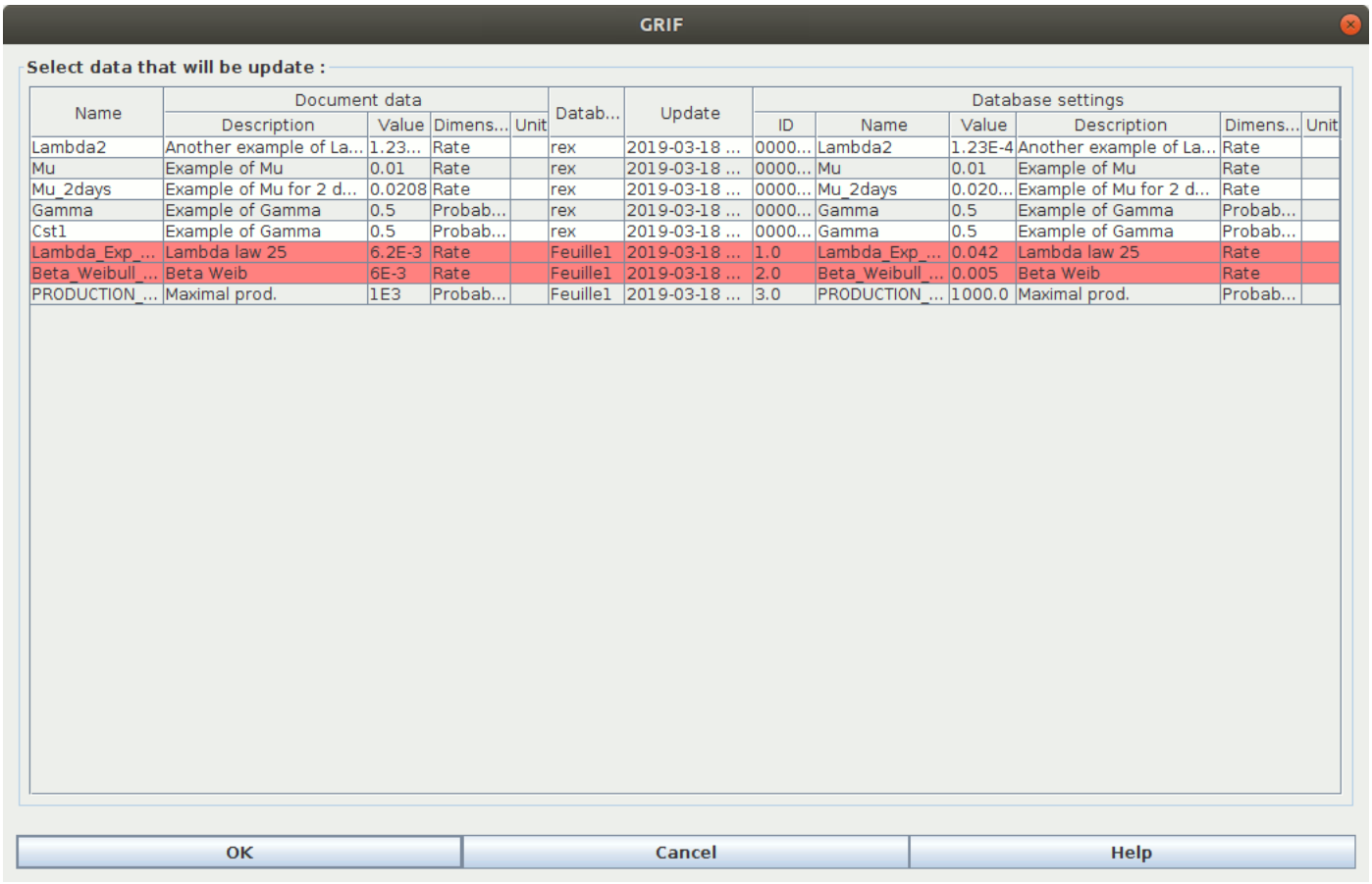


Select the parameters you want to import, and click on **OK**. The parameters are now created and imported in GRIF. The created parameters have the same names than the database's parameters, and the fields "Description" or "Dimension" are identical of those found in the database.

It is important to underline that it is possible to manually create a parameter in GRIF, and then with its **Add-On** menu, assign the parameter's value of the connected database. This operation is detailed in this link.

13.4. Update of the parameters from the database

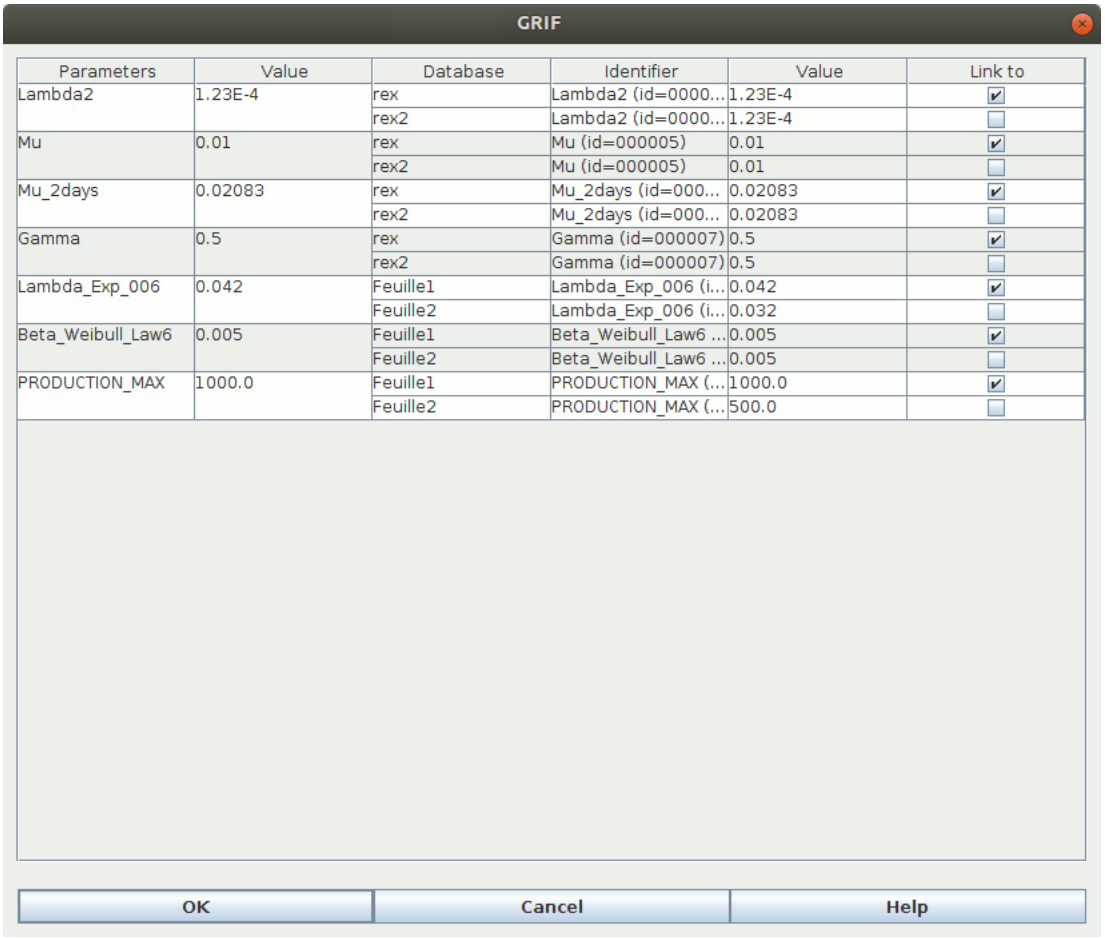
When an user, who has updated some of his data in his database, wants to have these modifications done on his parameters in GRIF too, he can then use the update action, from the **Data and Computations -> Parameters database -> Update from database ...** menu:



This window shows the parameters in GRIF which are connected to parameters from the databases. The red lines correspond to data which have been modified in the database. If the user wants to update some of his parameters in GRIF, he must select the lines of the wanted parameters, and then press the **OK** button. The parameters are now updated.

13.5. Rebuild of the links to the database

It is possible to modify an existing parameter's connection in GRIF, by changing the database of its associated parameter. However the parameter can only connect to the parameters with the same name. This action is available by the **Data and Computations -> Parameters database -> Rebuild links to the database** menu.



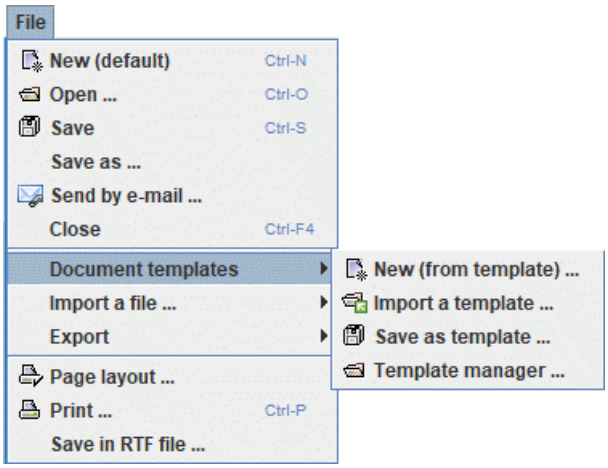
Here we can see the different parameters of the databases, which are imported in GRIF, and which are on multiples databases. So on the line of the parameters you want to rebuild the links, select the right database, and then validate your modifications by clicking on **OK**. GRIF then update the values of the parameters by rebuilding the links.

14. Save

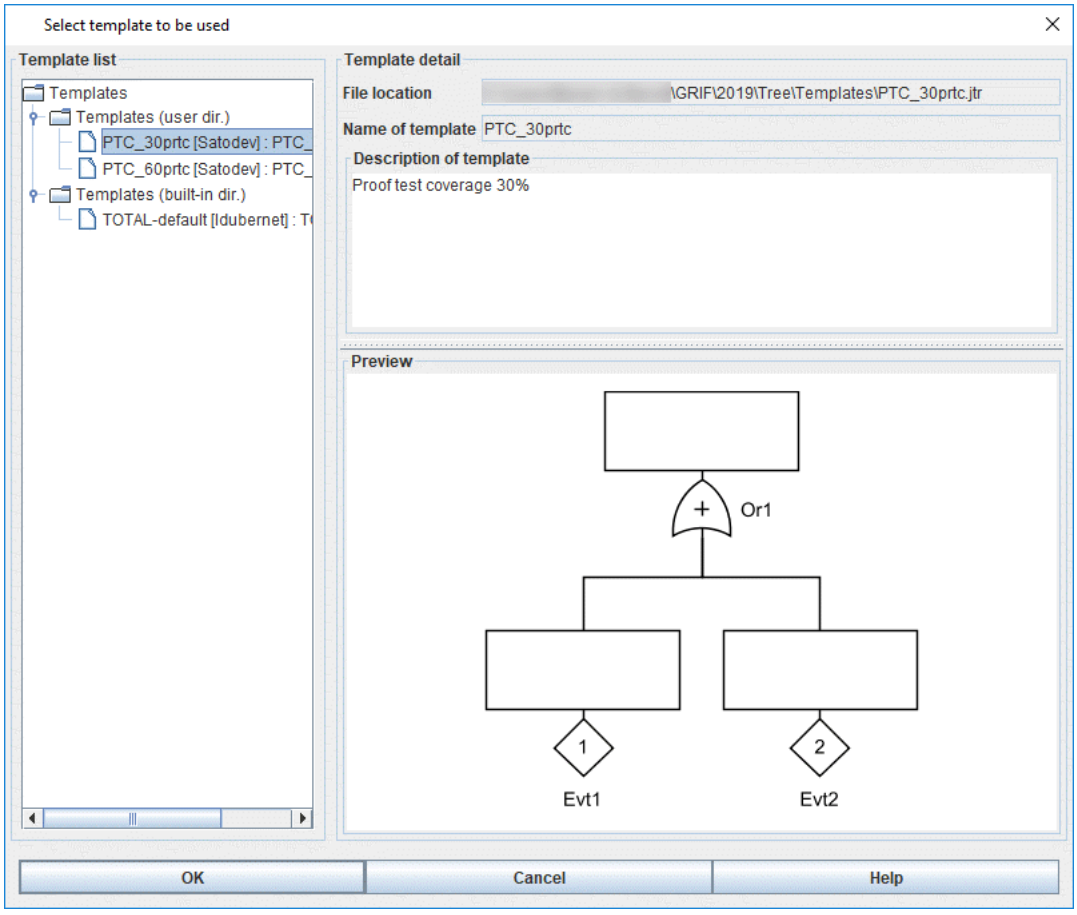
Here is a summary of all the data which can be saved from a same model.

14.1. Document template

It is possible to use an existing document as base to create a new document or as a part of a document. This functionality is accessible in **File - Document template** menu.

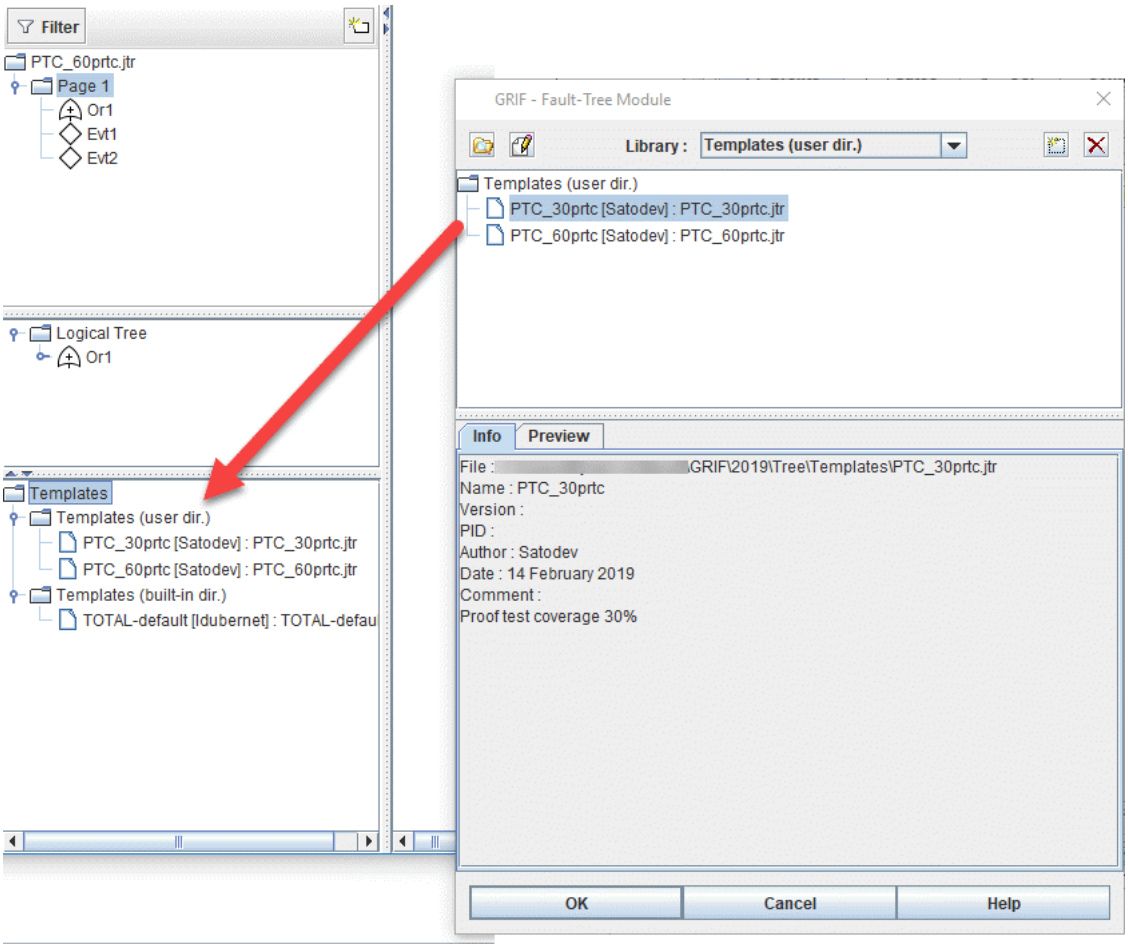


New (from template)... menu enables to open a new document and to initialize it with data from a model already build. A window appears to select the existing model.

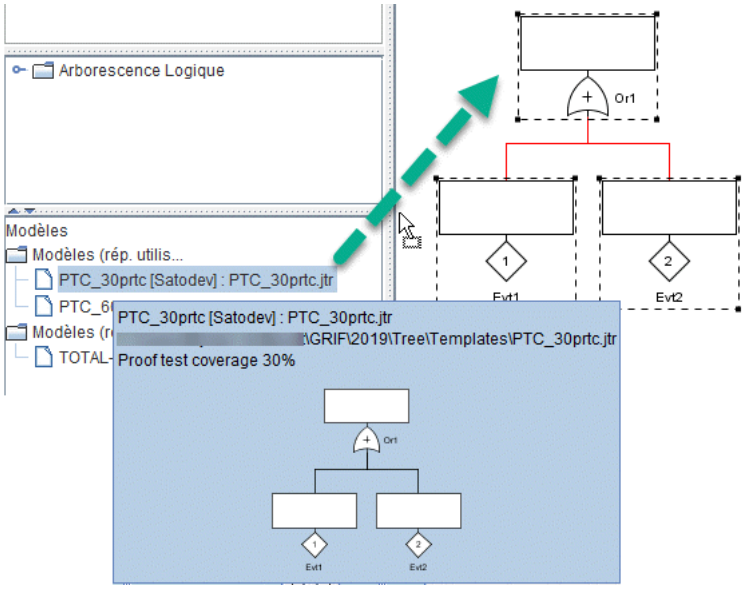


Import a template... menu enables to add to the current document data from a model already build.

Save as template enables to save the current document as template in the Template directory of the module. Once saved as a template, the document appears in the Template tree of the GRIF window as well as in the **Template Manager**.

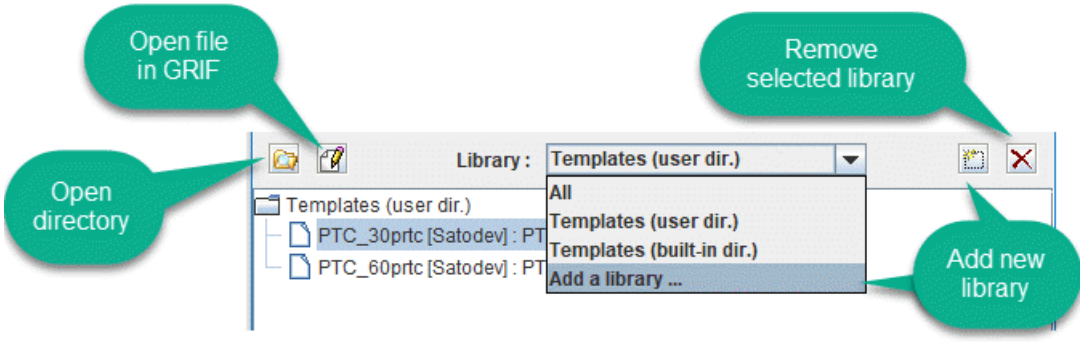


It is possible to create new files from this model using **New (from a template)...** action. A drag and drop to the templates from the input area enables to import the model quickly.



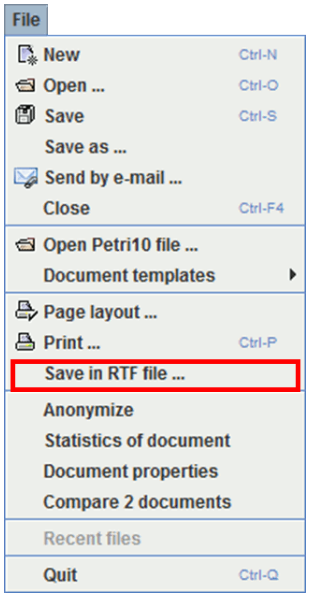
Save as default template menu enables to save the current document as default model in the module template directory. This model will also be the default model of the module. It will be used as base for creating a new document when **File - New (default)** action is used.

Template manager menu opens a window to manage the template of the document. New document libraries can be added/deleted. To add a new library it is necessary to select a directory of the file system. The tool analyzes the documents in this directory and builds a library that can be used by GRIF based on the compatible documents found.



14.2. RTF File

A model can also be saved in RTF format. This enables the saved model to be reloaded in WORD to insert the graphical part of the model in any document. To do this, go into menu **File - Save in RTF file...**.



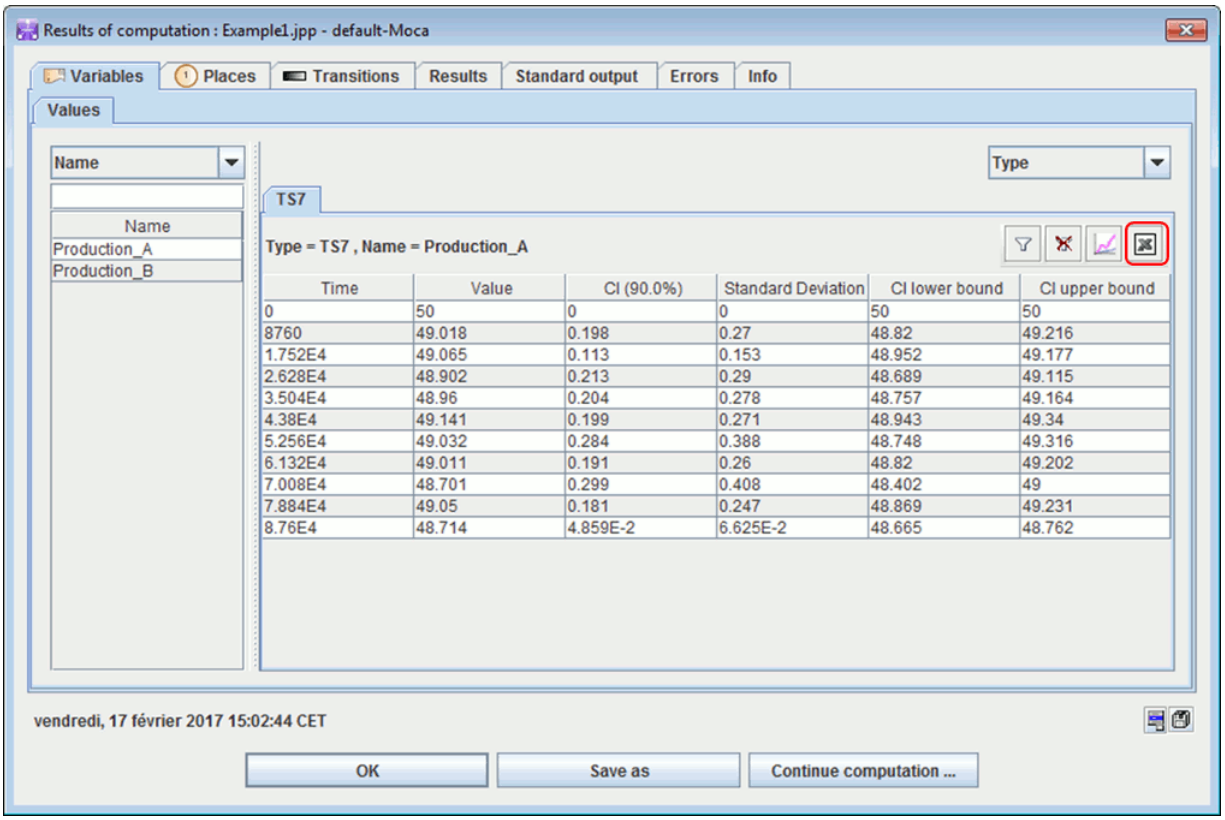
There is another way to insert model in a report.

Select the part of the model, copy it, and paste it in Microsoft WORD or other software.

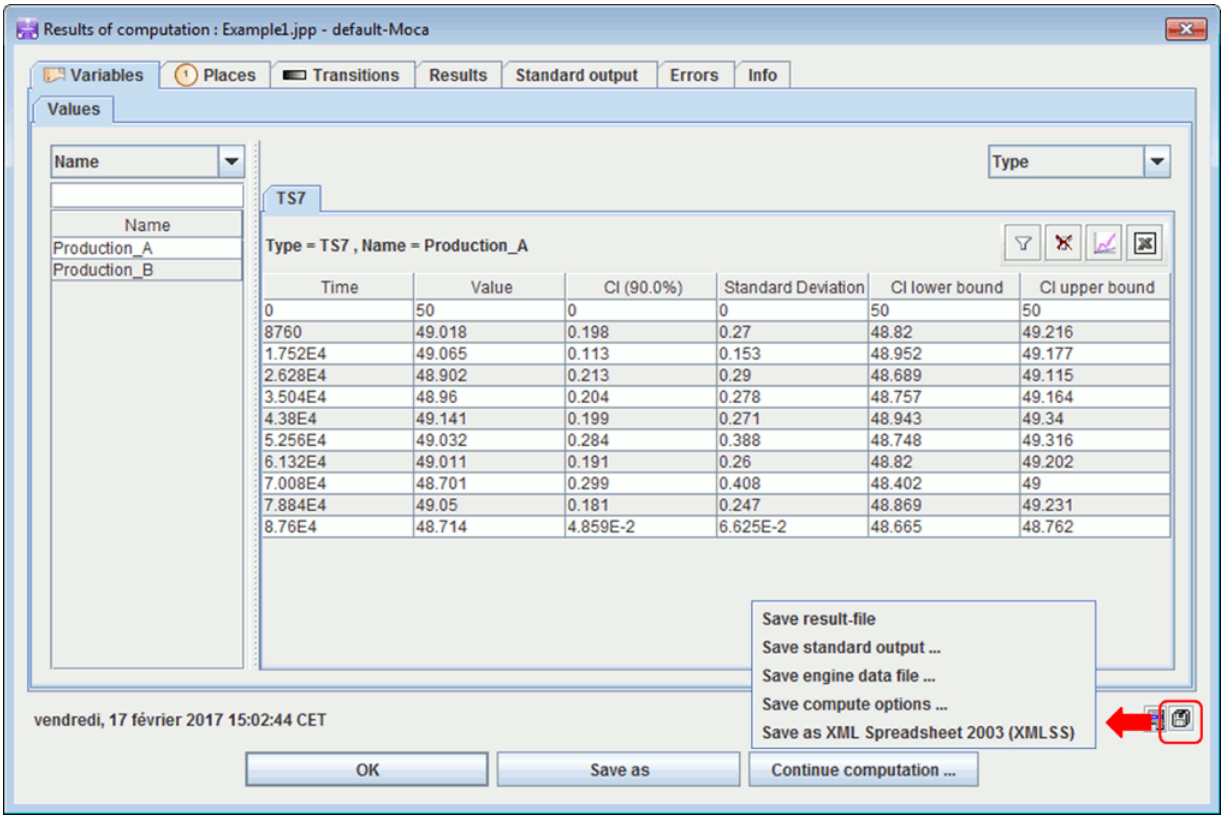
14.3. Results

Computation results can be saved in different formats:

1. Export of a table in particular in .csv format:



2. Export entire set of results:

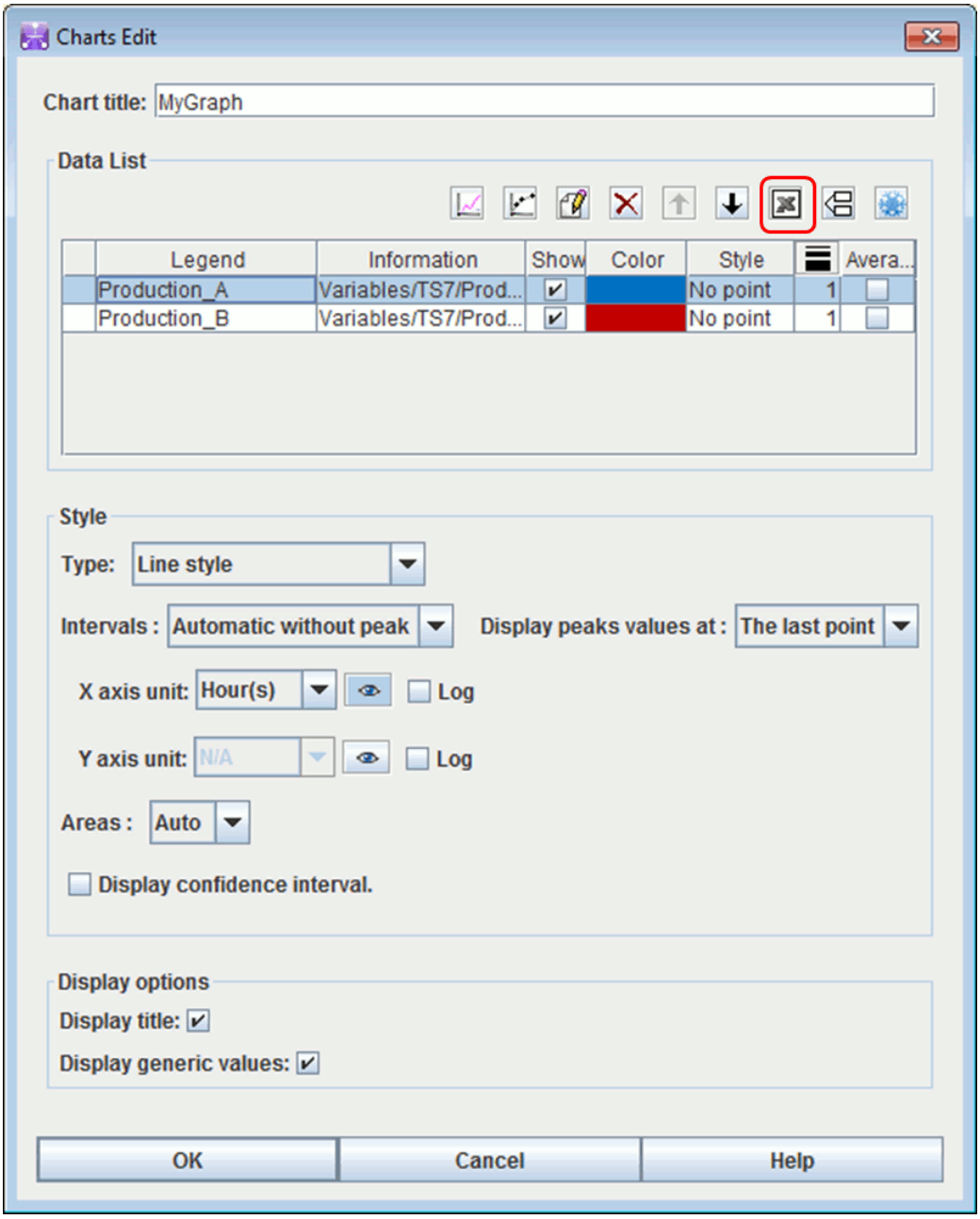


- **Save result file**: saves the contents of the Results tab (.xml format).
- **Advanced report**: generates reports using style sheets.
- **Save standard output**: saves the contents of the Info tab (.txt format).

- **Save engine data file**: saves the data file sent to the computation engine (.txt format).
- **Save as XML spreadsheet 2003 (XMLSS)**: saves all the results tables in an XML format compatible with Microsoft(r) Excel 2003 and later versions.

14.4. Curves

For each curve drawn, the points which have been computed in CSV format can be saved. This list of points can then be used to draw new curves or to perform further computations.



14.5. Tables

The different data table (places, transition, variables, parameters,etc...) can be saved.

For this, it is enough to click on the icon in the table.

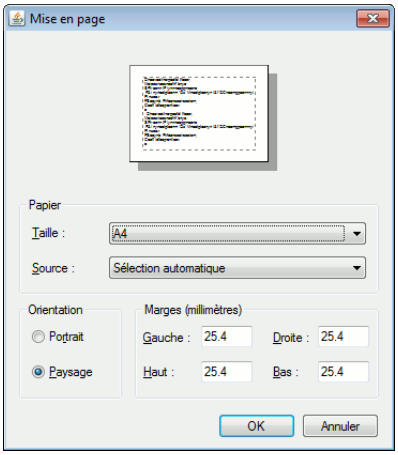
15. Printing

For printing, you have several commands at your disposal in the **File** menu **File**:

- **Page setup**
- **Print**
- **Save in RTF file**

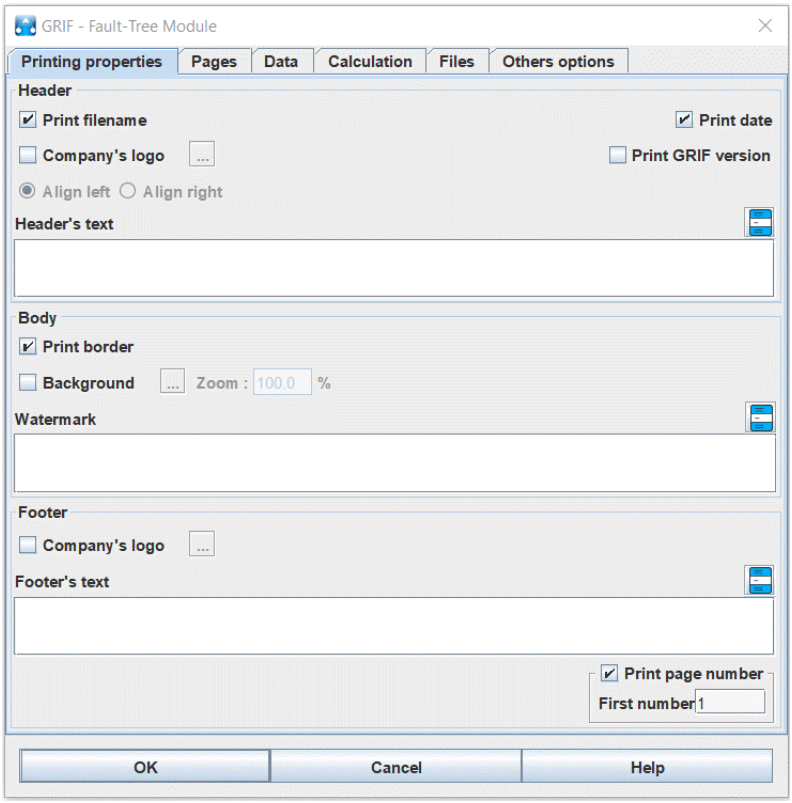
15.1. Page setup

The **Page setup** function enables you to choose the page orientation, the size of the margins, etc.



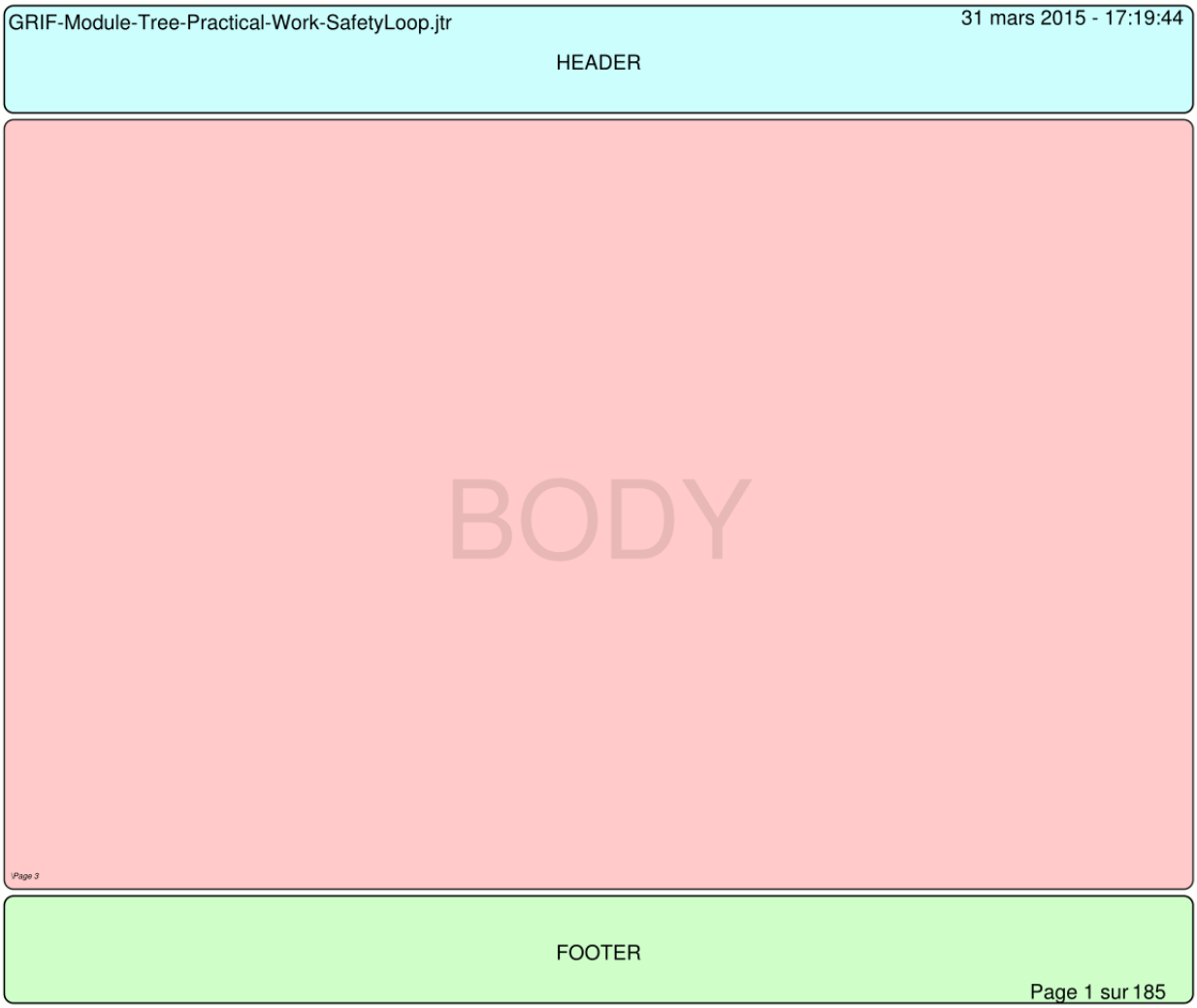
15.2. Print

The **Print** function enables you to export .pdf document pages. Graphics are exports in a vector graphics format in order to scale it without deterioration. All data table and the results of calculations can be also exported. Here is the window of configuration of the printing:

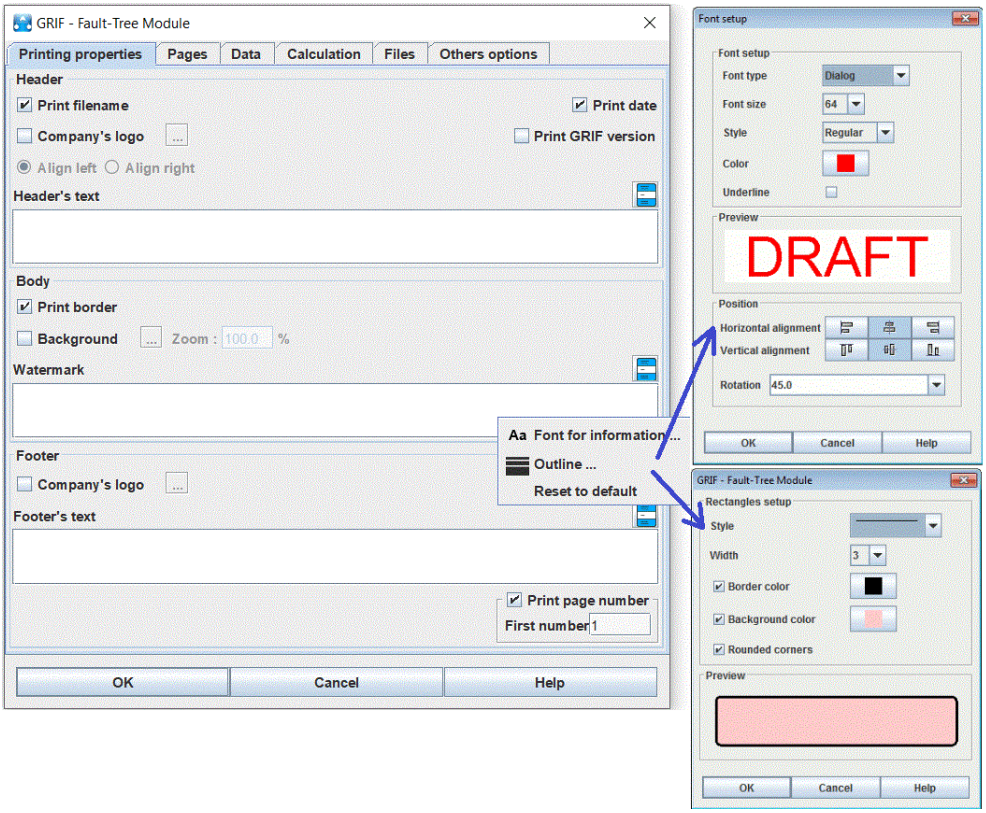


1. Printing properties

Printing properties tab gives the possibility of configuring what will be visible on all the printed pages. This tab contains three parts: the header, the body and footers. Below here is an example which illustrates the various zones on the PDF:



Each of three text fields is configurable. The user can so modify the police, the style, the color and the positioning of the text in the zone. He can also configure the zone itself by allocating a background color and a frame. A click-right on the text field shows a contextual menu: **Display settings**

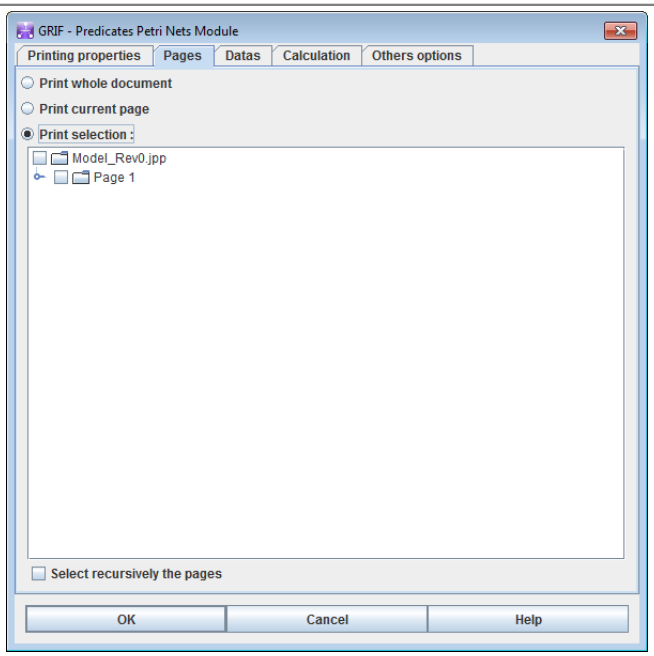


Several options are added:

- **Print file name** in the header at the left top.
- **Print date** in the header at the right top.
- **Print a border in the graphic pages** enables to bound visually the pages of graphs besides of the page of printing. Warning: this frame can be in conflict with a possible frame defines in the zone of body of page
- **Watermark** enables to seize an image to be printed on a background with colors which will be limited. The image can go out of the zone of body of page
- **Print number of page** in the footer in the lower right. The first number can be chosen to start not at 1.

2. Pages

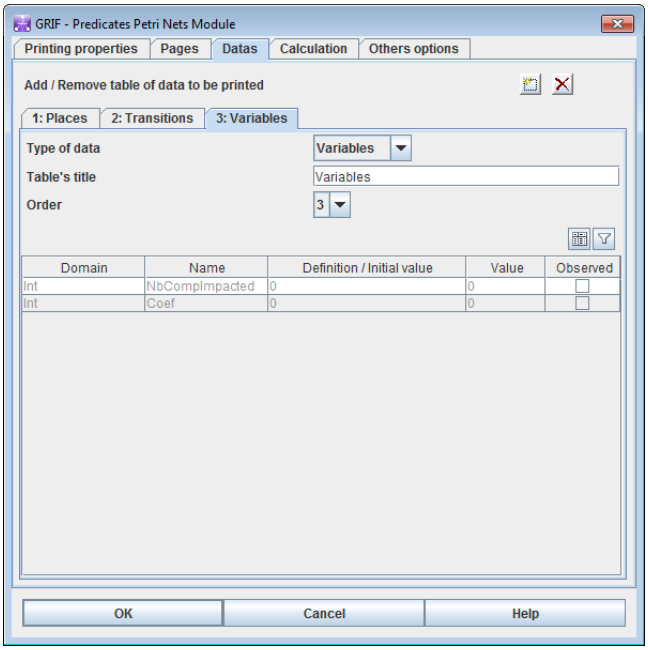
Pages tab enables to select the page will be printed.



- **Print all the page** : all the pages will be printed
- **Print current page** :only the current page will be printed.
- **Print selection** : select pages and groups will be printed. **Select recursively the pages** option is a help in order to select quickly sub-pages.

3. Data

Data tab enables to select date will be printed.

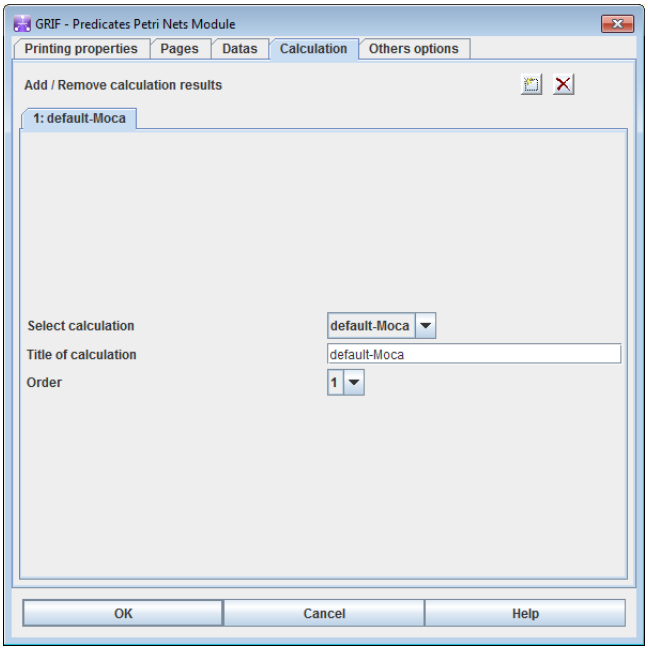


Each sub-tab will represent a table in the PDF document. The title and the print order of these tables can be modified.

By default all the types of data are represented (one by sub-tab). The upper right buttons allow to add it or to delete it. To note that we can print several times of the same data type in different tables. The data table can be filter using the usual filter. The Column manager can be also used.

4. Calculation

Calculation enables to select the calculation will be printed.

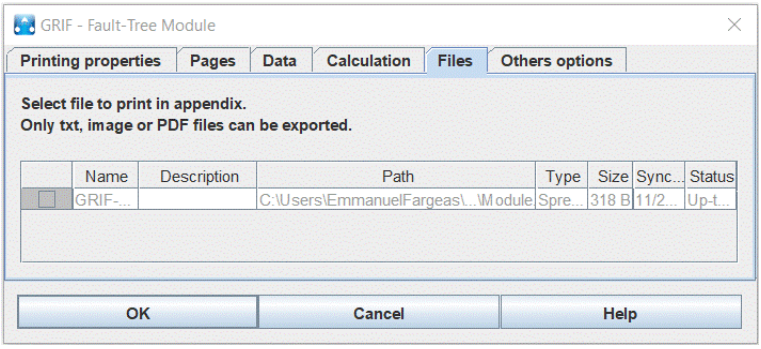


Each sub-tab will represent a part containing the various tables present in the results of the calculation. This part has a modifiable title and order which is the order of printing of the various results.

By default all the calculation results are represented (one by sub-tab). The upper right buttons allow to add to it or to delete it.

5. Files

Files enables to select the files related to the document to print in appendix. Only txt, image or PDF files can be exported.



6. Others options

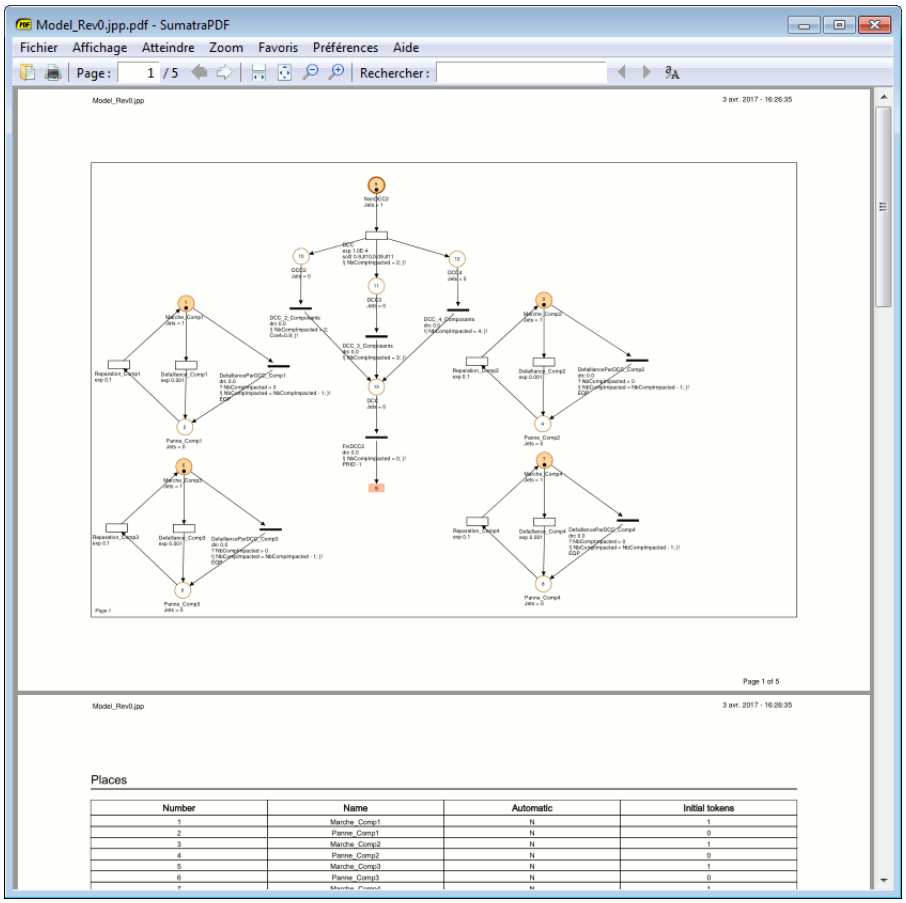
Others options .

- **print table of contents** : Allows to print the table of content of document, very useful for big documents.
- **Print modification tracking** : Allows, if the modification tracking optuion is selected, to print comment, author and date of modifications for each saving.
- **Print index of graphic pages** : Allows to print an index of graphic pages, very useful for big documents.
-
-

-
-
-

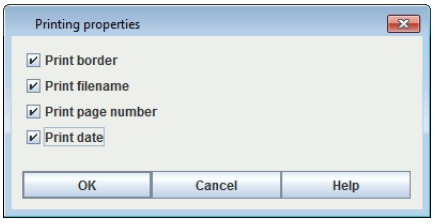
7. Example

To give an overview of a PDF printing:

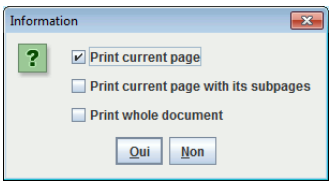


15.3. Save in RTF file

The **Save in RTF file...** function initially gives access to a window called **Printing properties** . Then to another called **Information** . And thirdly, a window is displayed allowing you to choose the folder in which the RTF file is to be saved.



When you select the **Save in RTF file** function, the first box to appear is that shown above. You can then select your preference: Print border, Print filename, Print page number and/or Print date.



Secondly, an **Information** window appears. It enables you to indicate whether you wish to print the current view, print the current page or print the whole document.

16. Options of GRIF - Petri Nets with predicates

Tools - Application Options menu opens a window containing the following tabs:

16.1. Options

Options tab enables to tune application behavior :

- **Save the options of the current document as default options in the application** : Save options of current doc as application default options.
- **The application manages the default options of the documents, apply the default options to the current document** : Apply -Application options- to current document.
- **Delay of automatic document saving (in minutes)** : Delay of automatic document saving (in minutes). A null value disables automatic saving.
- **Number of undo** : Specifies number of possible undo/redo.
- **Number of recent files** : Specifies number of files in recent files list.
- **Window display** : Enables separate tables (external) or linked tables (internal).
- **Columns to be resized in tables** : Enables to specify the columns on which space will be taken for resizing.
- **Ask for confirmation before deletion outside the input area** : When deleting an element in the graphic tree or in the table date, a dialog box will be displayed.
- **Manage new names to avoid name conflict** : Tries to avoid name conflict, creating new objects whose name is unique (when pasting for example).
- **Synchronize view with tables** : Select objects in tables (on the right) when they are selected in view.
- **Synchronize view with explorer** : Select objects in explorer (on the left) when they are selected in view.
- **Ask for confirmation if closing with close button** : When closing with the button at the top-right of a dialog box, the software will ask for a confirmation. Use OK or CANCEL buttons if you don't want to confirm closing.
- **Modification tracking when saving** : When saving, if tacking is activated, you can add a comment about modifications made on the document.
- **Additional syntax checking before computation** : Add a syntax/semantic checking of variables before computations and step by step simulation.
- **Ask for configuration of observed variables** : Ask for computation setup when variable becomes observed.

16.2. Executables

Executables tab enables to specify path to external executables :

- **Mail client** : Enable you to set the mail client to use
- **Automatically open PDF files** : Specifies if PDF reports must be opened with generation.
- **Moca-RPC path** : Specifies path of Moca 10.

16.3. Graphics

Graphics tab enables to modify GUI look :

- **Use Windows look and feel** : Use the look and feel of your operating system instead of java look and feel (GRIF restart is needed).
- **Element Zoom** : Changes graphics size.
- **Filling and outline for dynamic fields** : Object outline configuration (line color, line width, background color, ...).
- **Font for dynamic fields** : Enables font configuration (color, size, italic ...) for information that are displayed under objects.
- **Shape filling and outline for commentaries** : Object outline configuration (line color, line width, background color, ...).
- **Font for commentaries** : Enables font configuration (color, size, italic ...) for information that are displayed under objects.

- **Shape filling and outline for groups** : Object outline configuration (line color, line width, background color, ...).
- **Font for groups** : Enables font configuration (color, size, italic ...) for information that are displayed under objects.
- **Activate cross hair** : Activate cross hair which enables object alignment.
- **Activate smoothing for texts** : Activate anti-aliasing (smoothing) for texts, it can slow the display.
- **Activate smoothing for images** : Activate anti-aliasing (smoothing) for images, it can slow the display.
- **Activate tooltips** : Activate tooltip-system.
- **Configuration of automatic layout** : Let you configure the direction of automatic layout done with F7.
- **Part that will be laid out** : Let you define if the the layout is made on upstream or downstream part when you press F7.
- **Draw places and transitions according to IEC 62551** : Draw places and transitions according IEC 62551. Transitions will be drawn depending on their law.

16.4. Digital format

Digital format tab enables to customize digits display :

- **Display of parameters** : Specifies the display of parameters (number of digits, ...).

16.5. Computations / Results

Computations / Results :

- **Light Batch** : Deletes files used for each computation of batch computations, it decreases memory/disk use.
- **Preferred frequency unit** : Unit that will be used for displaying result which dimension is "frequency" in - main view, - data tables, - and some result synthesis.If no unit is displayed (especially in detailed results) the unit is (h-1).
- **Preferred duration unit** : Unit that will be used for displaying result which dimension is "duration" in - main view, - data tables, - and some result synthesis.If no unit is displayed (especially in detailed results) the unit is (h).

16.6. Places

Places tab defines places display options. :

- **Font set-up** : Enables font configuration (color, size, italic ...) for information that are displayed under objects.
- **Display name** : Enables to display name or not
- **Display number** : Enables to display number or not.
- **Display description** : Enables to display description or not.
- **Display number of tokens** : Enables to display number of tokens or not.
- **Display attributes** : Enables to display attributes or not.
- **Display default value for attributes** : It displays the default value of attribute for which no value is defined.
- **Display name on shortcuts** : Enables to display the name of reference place on each shortcut.
- **Additional information** : Enables to display additional information (results of computation)

16.7. Transitions

Transitions tab defines Transitions display options :

- **Font set-up** : Enables font configuration (color, size, italic ...) for information that are displayed under objects.
- **Display name** : Enables to display name or not
- **Display number** : Enables to display number or not.
- **Display description** : Enables to display description or not.
- **Show HISTO characteristics** : Enables to display HST flag of transition or not.
- **Display firing options (PRIO, EQP ...)** : Enables to display firing options of transitions.
- **Display guards** : Enables to display guards of transition or not.
- **Display assignment** : Enables to display assignments of transition or not.
- **Display law** : Enables to display law of transition or not.

- **Use a default law** : Enables to use a specific law as default law.
- **Display attributes** : Enables to display attributes or not.
- **Display default value for attributes** : It displays the default value of attribute for which no value is defined.
- **Additional information** : Enables to display additional information (results of computation)
- **Default law** : Default law that will be used when creating objects.
- **Maximum number of characters by line** : Specifies maximum number of characters per line.
- **Lexical check of transitions before computing** : Activates lexical check of transitions before each computation.
- **Check transition conflict** : Checks that there is no transition firing conflict : 2 transitions with the same upstream place and exactly the same guard (same string).
- **Null transition rate produces a warning** : Displays a warning message if a transition rate is equal to 0.

16.8. Arcs

Arcs tab defines arcs display options :

- **Link arrow width** : Specifies arrow width.
- **Link arrow height** : Specifies arrow height.
- **Show arrows links to the foreground** : Displays the direction of links on the foreground. Otherwise, the arrows will be drawn behind.
- **Display arc weight** : Enables to display arcs weight or not.

16.9. Local data

Local data tabs enables to change local data display :

- **Font set-up** : Enables font configuration (color, size, italic ...) for information that are displayed under objects.
- **Display data domain** : Display domain of definition.
- **Display name** : Display data name.
- **Display description** : Display data description.
- **Display data definition** : Display variable definition.
- **Display data value** : Display data value.

16.10. Simulation

Simulation tab enables to set simulation up :

- **Automatic firing of dirac 0** : Automatic firing of transition with Dirac 0 law.
- **Automatic firing when simulation starts** : Automatic firing of transition when simulation starts.
- **Transitions fireable before loop detection** : Specifies maximum number of fire before loop detection.
- **Limit duration for "fast forward" (s)** : Specifies the number of seconds before the detection of a loop during the "fast forward".
- **Graphical Rendering of simulation** : Activate or not graphical rendering of simulation or each modification.
- **Keep history during "fast forward"** : It enables history-saving of triggered transitions, the ones automatically triggered when using Go until time/transition functions
- **Constant time interval between two automatic fires** : When using automatic forward during step by step simulation, the (user) time between two transition fires will be constant. If this option is not checked, time will be proportional to the simulated time.
- **Delay between transition fire (ms)** : Delay between two automatic fire.
- **Simulated time / user time coefficient** : Delay between two automatic fire.
- **Simulation trace** : Enable simulation trace.
- **Path to trace file** : File where simulation is saved.
- **Follow scheduler order for deterministic transitions** : Only the first deterministic transition of scheduler can be fired.
- **Choose delay of fired transitions** : Displays a dialog box to choose delay of fired transition.
- **Uncertainty propagation for interactive simulation** : If not checked, the value of variable with uncertainty will be the average. For example, 1.5 for a unif(1,2).
- **Display histogram (transition)** : Enables to choice type of histogram for transitions.
- **Parameter (nb classes, nb and size of steps ...)** : Specifies parameter for transition histograms.

16.11. Curves

Charts tab enables to change charts drawing :

- **Set graphics borders** : Add borders to charts.
- **Set generic values borders** : Add borders to generic values under charts.
- **Display grid** : Display grid on curves area.
- **Display legends** : Display legends under curves.
- **Drawing zone transparency** : Activate curves area transparency.
- **Graphic transparency** : Activate charts transparency.
- **Title size** : Specifies charts title font size.
- **Generic values size** : Specifies generic values font size.
- **Point size** : Specifies point size on curves.
- **Coordinates size** : Specifies coordinates font size.
- **Legend size** : Specifies legends font size.

17. Annex

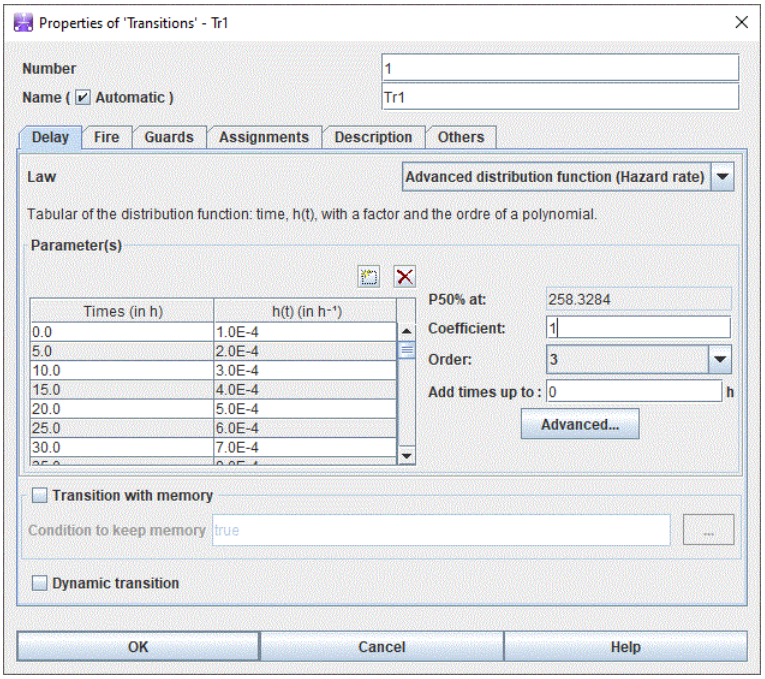
17.1. Hazard Rate h(t) Utils

The Hazard Rate law (denoted h(t)) enables to create a law from the failure rates of a component. This law is formed by multiples pairs, a pair representing a time t and its associated failure rate at this time t. An utility for the computation of Hazard Rate Laws is given by GRIF, which give various functionalities, listed below:

- Computation of the H(t) values (integrals of the h(t) values)
- Computation of the F(t) values (values of 1 - exp(-H(t)))
- Computation of the F(t) values with a coefficient specified (denoted qF(t))
- Computation of the qH(t) values, calculated from the qF(t) values (values of -log(1-F(t))), and the possibility to add times up to a specified date.
- Computation of the qh(t) values, from the qH(t) values (derivate of the values of qH(t))
- Modelization of the points from the laws quoted above in graphics, to allow to compare the values before and after the multiplication of these values by a coefficient.
- Computation of 5 polynomial of orders 1,2,3,4,5 representing the values of qH(t), and their visualization.
- Differences between the polynomials, displaying the least squares and the correlation between the qH(t) values and the values computed from the computed polynomial, for each order.

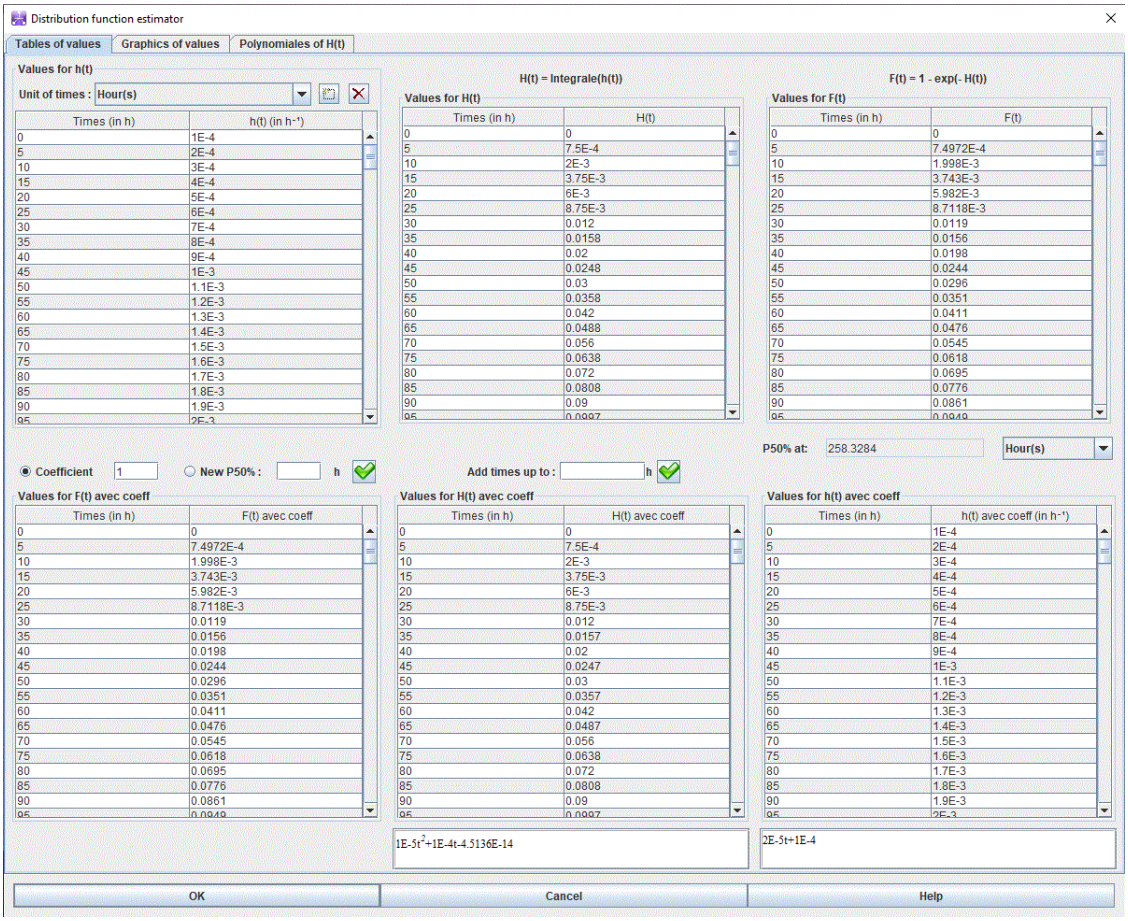
The main purpose of this utility, besides of the visualization of the h(t) law, is to create a new law qh(t) from the h(t) law, but with a coefficient assigned, via the process h(t) => H(t) => F(t) => qF(t) => qH(t) => qh(t).

This utility can be found in the "Advanced distribution function" law's panel, which from his h(t) values, will create a "Cumulative distribution function", with as parameters the qF(t) values (which can be seen in the utility).



This utility is made of three tabs, which are going to be described to guide on how to use this utility.

17.1.1. Tables tab



The first tab list the values tables. The only editable table is the h(t) table, where the date must be directly entered. The others tables are refreshed automatically, depending on the data entered by the user in the h(t) table.

17.1.1.1. Table of the values of h(t)

The table consist of two columns: one representing the times, the other the failure rates. The user can specify the times' unit of these values, by selecting the correct time in the menu above the table. Two buttons allow to add a row of values, or to suppress one or more. The user can, when doing a paste action (Ctrl+v) on a time cell, import multiples times and failure rates, but he needs to be sure his values are correctly organised: the times left, and the failure rates right.

17.1.1.2. P50%

The field P50% enables the user to know at what time the values of F(t) exceeds 50%. The user can inform the times unit of the P50% he wants, and this unit will be used by the new P50% and the possible addition of new times, detailed below.

17.1.1.3. Coefficient or new P50%

The coefficient can be specified in two different ways: the first way the most explicit consist of entering the wanted coefficient in the field "Coefficient". The second one allow to enter a new wanted P50%, and it will create a coefficient, equals to the new P50% divided by the ancient one.

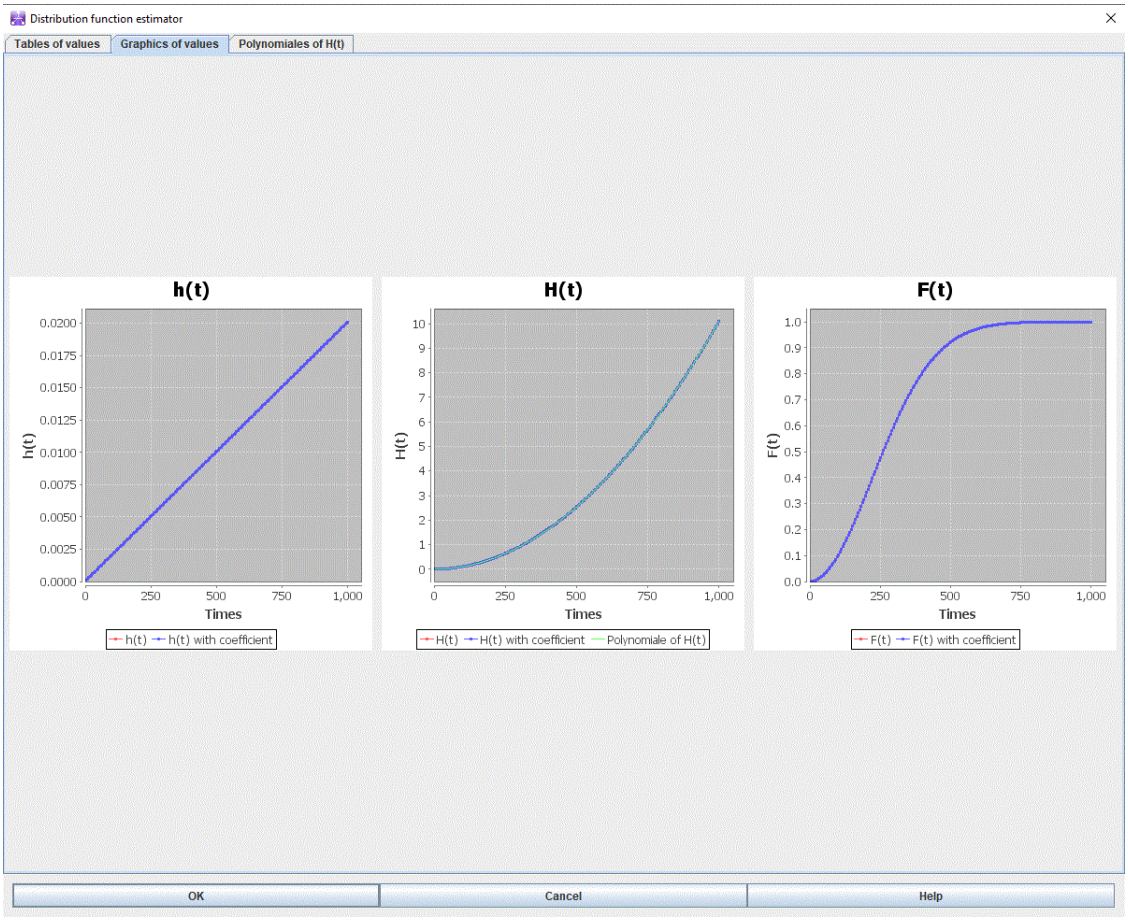
17.1.1.4. Addition of times in the table of qH(t) values

The user can entered a new time t, to fill the table of new rows of values, up to the informed new time. The medium interval between the ancient times are calculated, and the new times to add are calculated according to the last time and the value of this interval, until reaching the new time t.

17.1.1.5. Polynomial's fields

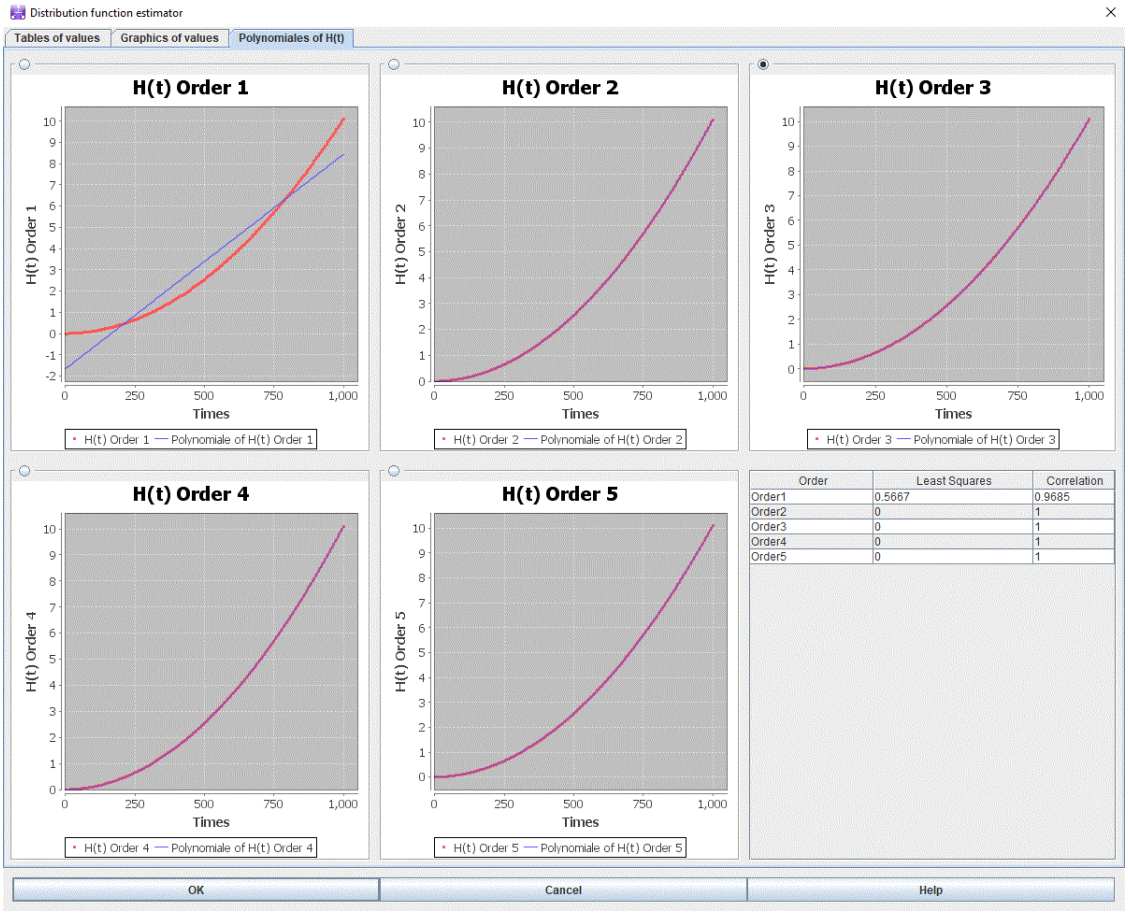
Two fields, located below the tables of qH(t) and qh(t) values, display the computed polynomials which represent the values of these tables. The table qh(t) representing the values of the derivate of the qH(t) polynomial, the polynomial below the qh(t) table is the derivate of the qH(t) polynomial. By default, the order of the polynomial figuring below the qH(t) table is 3 (so 2 for the polynomial below the qh(t) table), but the user can choose the order manually, by selecting one of the button figuring in the "Polynomials" tab.

17.1.2. Points tab



This tab enables to see the differences which occured during the multiplication of the times by the coefficient. The tab contains three graphics, each one of them representing the values of one of the three laws seen previously, as well as its equivalent multiplied by a coefficient. The first one display the values of h(t) and qh(t), the second one H(t) and qH(t), and the last one F(t) and qF(t). The second graphic has a feature, which displays the values of the computed polynomial, to see if the H(t) values and the computed polynomial values are adequate. This tab must therefore be opened after all of the date has been entered (all h(t) values, coeff, ...) in the "Tables" tab.

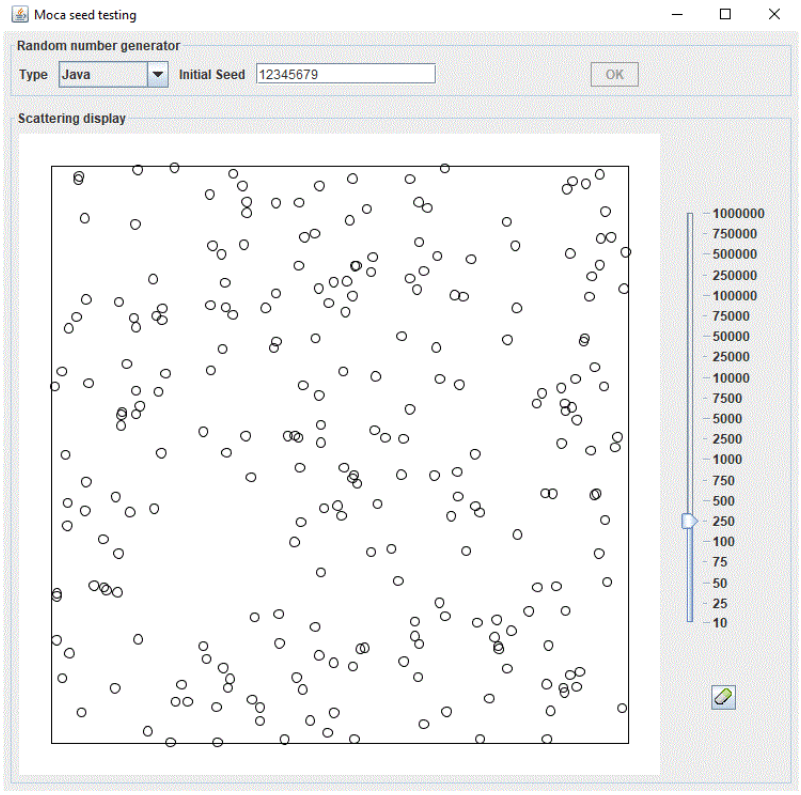
17.1.3. Polynomials tab



The last tab displays all the computed polynomials (depending on the values of the qH(t) tab), and the table displaying the least squares and the correlation between the values of qH(t) and the values of the computed polynomial. The user can select one of the buttons located above the graphics, to specify which polynomial is displayed in the polynomial field located in the "Table" tab, and in the graphic of the H(t) and qH(t) values located in the "Points" tab.

17.2. Moca seed testing utilitary

The Moca seed testing utilitary enables you to test a random seed, to see if the generated numbers have a good dispersion. To access to this utility, click on the menu **Utils**, then **Moca Seed Testing**.



The first part of this utilitary enables you to enter the type of the generator, and the initial random seed.

- The first is the JAVA random generator.
- The second is the historical Moca random generator, which is used in the Moca version lower or equals to 13.
- The third is the Moca31Int generator, allowing you to make a number of stories below 2 billions.
- The forth is the Moca48Int, allowing you to make a number of stories higher than 2 billions, and which is inspired by the pseudo-random numbers generator defined by D. H. Lehmer and described by Donald E. Knuth in "The Art of Computer Programming", Volume 3: "Seminumerical Algorithms", section 3.2.1.

Once the initial random seed is entered, you can launch the computations, and it will fill the scattering table, in the second part of the utility. There is a ladder at the right of the table, to inform the numbers of generated numbers to displayed. It is now possible to judge by the scattering of the generated numbers if the seed generate well-scattered numbers, or not.