GRIF 2022

Module SIL

User manual

Version May 6th, 2022

Copyright © 2022 TotalEnergies

Abstract

This document is the user manual of SIL Module of *GRIF 2022*

Table of Contents

1. Prerequisites and Installation 6
1.1. Prerequisites 6
1.2. Installation of TotalEnergies version without installer 6
1.3. Installation of retail version with installer and demonstration version 6
1.4. Saving 6
1.5. Launching 6

2. Presentation 7
2.1. Introduction 7
2.2. Main window of the SIL module 7
2.3. Vertical toolbar 8

3. Configuration of architecture 9
3.1. Architecture definition 9
3.2. Use of graphical input zone 10
3.3. Voting for components of a channel 10
3.3.1. 1oo2S and 1oo2A 11
3.3.2. 2oo3S 12
3.3.3. 2oo3A 13
3.4. Configuration of channels of a part 13
3.5. Take Common Cause Failures into account 13
3.6. Constraints on SIF architecture 13
3.6.1. Definition 14
3.6.2. IEC 61508: Route 1H 14
3.6.3. IEC 61508: Route 2H 15
3.6.4. IEC 61508: Route 2H - Chap 7.4.4.3.2 15
3.6.5. IEC 61511 - Version 2016 15
3.6.6. IEC 61511 - Version 2016 - Chap 11.4.6 15
3.6.7. How to configure the standard that will be used for contraints 16

4. Configuration of components 17
4.1. Configuring the sensors 17
4.1.1. Existing component 18
4.1.2. Identification 18
4.1.3. Determined character of the component 19
4.1.4. Test 19
4.1.5. Instrument parameters 20
4.2. Configuring the solver 23
4.2.1. Solveur existant 23
4.2.2. Identification 23
4.2.3. Configuration 25
4.2.4. Instrument parameters 25
4.3. Configuring the actuators 26
4.3.1. Existing component 27
4.3.2. Identification 27
4.3.3. Determined character of the component 29
4.3.4. Test 29
4.3.5. Instrument parameters 29
4.4. Editing the parameters 32
4.5. Edition of data table 33

5. The parameters 34
5.1. Creation 34

6. Menus presentation 36
6.1. File 36
6.2. Edit 38
6.3. Tools 38
6.3.1. Charts model 39

6.4. Document 40
6.5. Data and Computations 40
6.5.1. Edit data tables 42
6.5.2. Parameters database 43
6.6. Add-ons 43
6.7. ? 43
6.7.1. Configuration 44

7. Attributes 46
7.1. Creation 46
7.2. Use of the attributes 47

8. Data Bases 48
8.1. Database of parameters 48
8.1.1. Format of the databases 48
8.1.2. Connect to a database 48
8.1.3. Import parameters from a connected database 52
8.1.4. Update of the parameters from the database 53
8.1.5. Rebuild of the links to the database 54
8.2. Bases of components 55
8.2.1. Export to xlsx 55
8.2.2. Connect to a component base 56
8.2.3. Disconnect 56
8.2.4. Base of component 56
8.2.5. Update the components 56
8.2.6. Convert the base to the new format of SIL 56

9. Computes 57
9.1. Launch PFD/PFH computation 57
9.2. Computations results 57
9.3. Compute manager 58

10. Multi-loop systems 60
10.1. Creation of sevaral instrumented loops 60
10.2. Import from a another document 61
10.3. Presentation 61
10.4. Input 62
10.5. Computations 62
10.6. Reports and results 63
10.7. PDF report and MS Excel report 64

11. Charts 65
11.1. Charts Edit window 65
11.2. Editing the curves 66

12. Risk matrices 68
12.1. Entering matrix acceptability levels 68
12.2. Entering risk matrix models 68
12.3. Entering coefficients 70
12.4. Risk matrix tool 70

13. Compare 2 documents 73

14. Zoom and page size 74

15. Hypothesis 75

16. Document properties / Track change / Images management 76

17. Files of the documents 79

18. Document template 80

19. Generating reports 83
19.1. Identification 83
19.2. Description 84
19.3. Result SIL 84

19.4. Spurious-trip 86
19.5. Image 86
19.6. PDF reports 87
19.7. Microsoft Excel XLSX file format export 89

20. Checksum 90

A. Configuration of Lambda computation method for CCF 91

B. Data Editing Tables 92

C. List of components 96

D. Options of GRIF - SIL 98

E. Law 101

F. Glossary 123

1. Prerequisites and Installation

This chapter describes the procedure to follow before the software can be used. Because an external computation engine is used, certain prerequisites are necessary.

1.1. Prerequisites

The minimum hardware requirement is a Pentium IV (or more) with 1 GB memory. Works under Window XP, Vista and 7

1.2. Installation of TotalEnergies version without installer

TotalEnergies' version of the GRIF software does not require an installation procedure. You must unzip the GRIF 201X-Module SIL.zip in the directory of your choice. The path of directory must not contain any special characters such as: {,},[,],(,),$,%, etc.

In the following chapters, we will assume you have unzipped the file in `C:\Programmes\Total\GRIF 201X-Module SIL\`

1.3. Installation of retail version with installer and demonstration version

The retail version of software is provided with a file whose name is GRIF 201X.zip . Unzip the file (on your desktop for exemple), ans launch GRIF-Install-Win32.exe. A window will guide you through installation step. If you haven't purchase GRIF, please select demo at the end of inttallation.

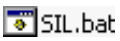
In the following chapters, we will assume you have install GRIF in `C:\Programmes\Total\GRIF 201X\`

1.4. Saving

Data generated by GRIF are saved in "USER" directory. With Windows XP it is `C:\Documents And Setting\UTILISATEUR`, with Windows Vista and Windows 7 it is `C:\Utilisateurs\UTILISATEUR`. The name USER is usually your name or your identification number with which you have openned your session on computer.

GRIF Module SIL saves its files in `USER/GRIF/SIL/Application` .

1.5. Launching

The software is now ready for use. To launch the SIL module, double-click on 

qui is in directory where GRIF has been installed. In retail version, you can also use the Start menu (Software/GRIF 201X).

2. Presentation

2.1. Introduction

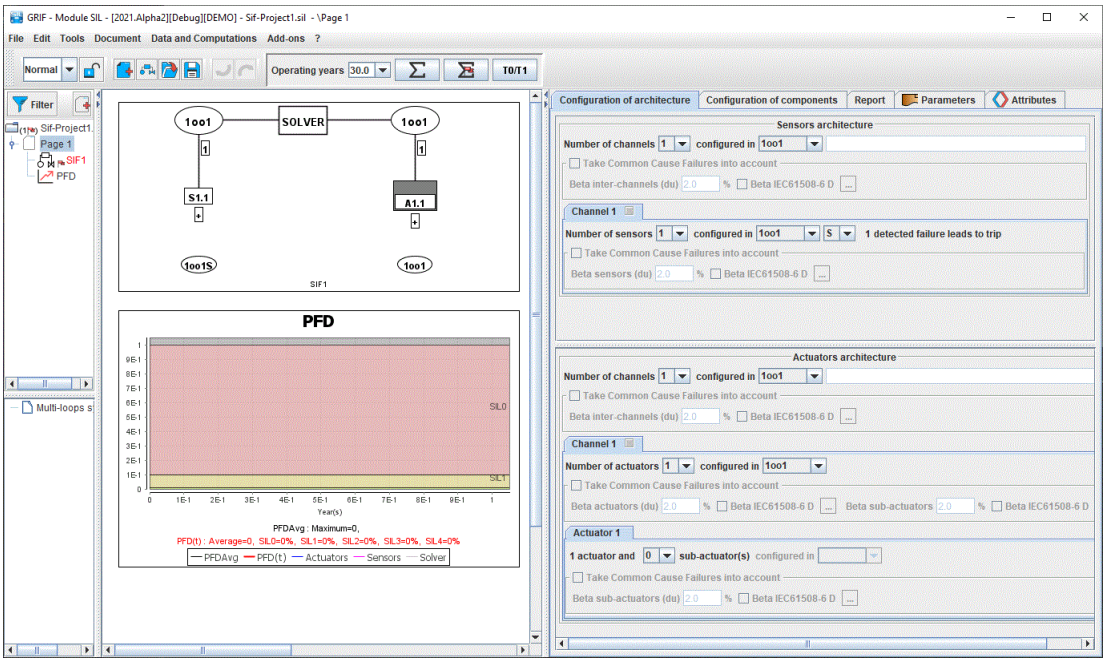
The SIL module in the *GRIF-Workshop* software platform enables instrument technicians in charge of architecture design or maintenance of SIS (Safety Instrumented System) to evaluate the SIL (Safety Integrity Level) of safety instrumented loops in line with standards IEC 61508 and 61511. The computations carried out are safety computations; the top event is a non-detected dangerous failure of the SIS (Safety Instrumented System) safety function.

The definitions and parameters used in this document are explained in the glossary (cf. Appendix F, *Glossary*)

2.2. Main window of the SIL module

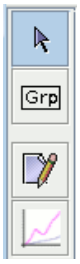
The main window is divided into several parts:

- **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. The **Operating duration** area lets you specify the operating duration in years, and launch computation.
- **Tool bar**: The tool bar (vertical) enables you to select the elements for modeling. By default, this tool bar is not displayed. In order to make it visible, check the box **Display graphical tools bar** in **Tools**.
- **Input zone**: A maximum amount of space has been left for the graphical input zone for creating the model. When module is launched, this zone contains a picture representing the architecture as well as an empty chart as no computations have yet been carried out.
- **Tree**: Graphical tree is between input zone and tool bar. It enables to walk through pages and groups of the document. It is not displayed by default.
- **Configuration window**: On the right of input zone, a window that contains **Configuration of architecture**, **Configuration of components**, **Report**, **Parameters** and **Attributes** enables you to configure the system.



2.3. Vertical toolbar

All the graphical symbols are shown on the vertical icon bar on the left of the data input screen. This toolbar is not visible by default, it can be displayed with the **Tools** menu.

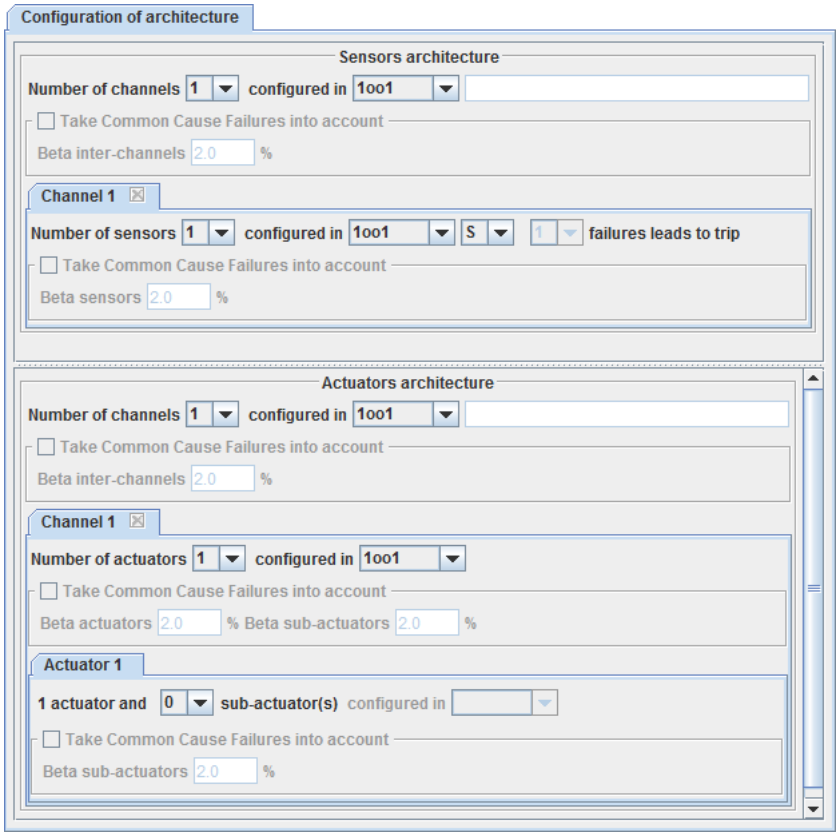


The vertical toolbar contains the following items:

- **Select** selects the desired elements.
- **Group** to create a new group. A group is a sub-page which can contains graphics elements.
- **Comment** to add text directly to the graphic.
- **Charts** to draw charts representing computations on the model.

3. Configuration of architecture

The **Configuration of architecture** tab enables to define architecture. Each modification made on this tab is visible on picture of architecture.



3.1. Architecture definition

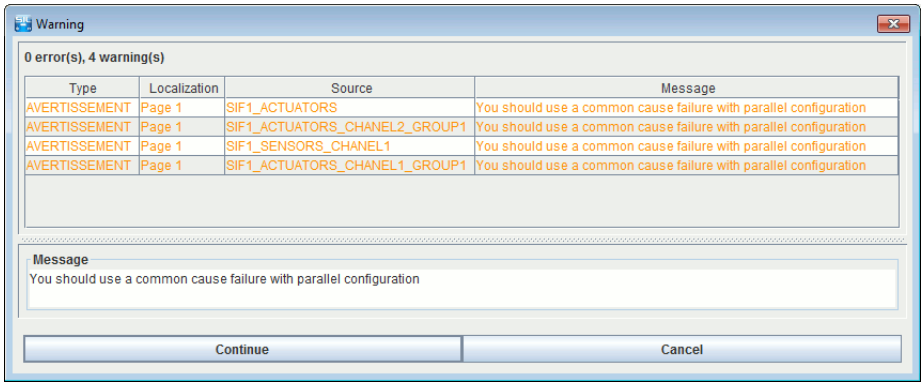
The top part of the tab: **Configuration of architecture** enables to define the configuration of sensor part. Possible choices are:

- The number of channels, up to 16, and the (logical) configuration between channels (cf. **Configuration des canaux**);
- Number of components in each channel, up to 24, and the configuration of these components in the channel;
- Taking Common Cause failure into account for all sensors (cf. **Configuration des DCC**);
- Taking Common Cause failure into account for sensors of a channel.

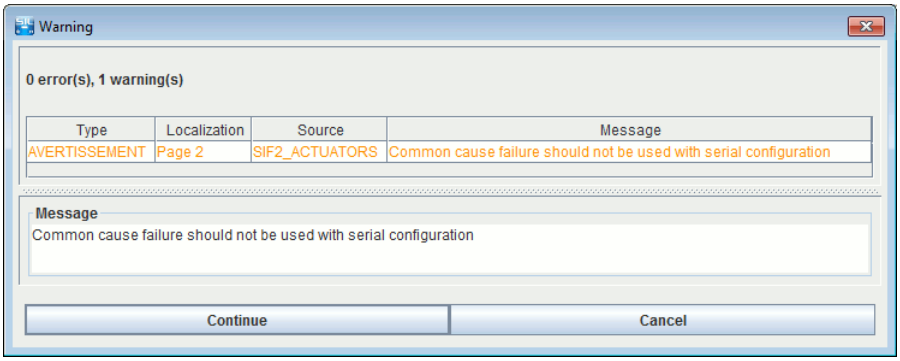
The bottom of **Configuration of architecture** tab enables to define the configuration of sensor part. Possible choices are:

- The number of channels, up to 24, and the (logical) configuration between channels (cf. **Configuration des canaux**).
- Number of actuators in each channel, up to 8, and the configuration of these components in the channel.
- For each actuator, the number of sub-actuators (0, 1, or 2), and the configuration of these sub-actuators.
- Taking Common Cause failure into account for all actuators. (cf. **Configuration des DCC**)
- Taking Common Cause failure into account for actuators and sub-actuator of the channel.
- Taking Common Cause failure into account for sub-actuators of the actuator.

In case of parallel and redundant architecture, a **warning** appears if user does not indicate common cause failures.



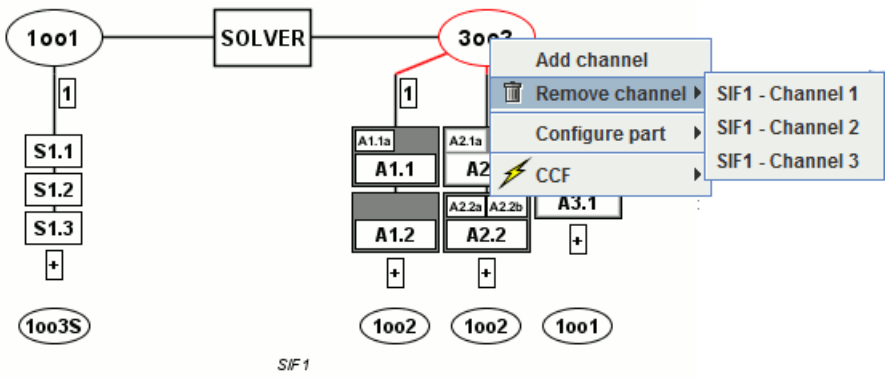
In case of serie architecture, a **avertissement** appears if user indicates common cause failures.



3.2. Use of graphical input zone

It is possible to remove, add a channel or modify redundancy of channel using graphical input zone of safety instrumented loops. A right click on the ellipse makes it possible to add or remove a channel. In this case, the last channel will be removed.

A right click on channel number removes the specific channel.



3.3. Voting for components of a channel

Usually, for each channel, a MooN voting means that you need M components at threshold (detecting problem) to trip (put the system under control in a safe mode). For sensors, the SIL module distinguish MooNS (Safety) and MooNA(Availability) voting.

- **Vote with "S" type architecture**: the invalidity of the sensor triggers the safety system (Safe).

- **Vote with "A" type architecture**: the invalidity of the sensor triggers no action other than an alarm (availability). The solver logic is modified, excluding sensors with detected failure. In this case, we define a number (X) of detected failure from which the channel trips. This number (X) is fixed by default for TotalEnergies (but can be modified in M configuration):
  - 3 if 3 components or more
  - 2 if 2 components
  - 1 if 1 component

- **Vote with "M" type architecture**: It is exactly the same definition as type "A". But X (the number of detected failure leading to trip) can be modified by users.

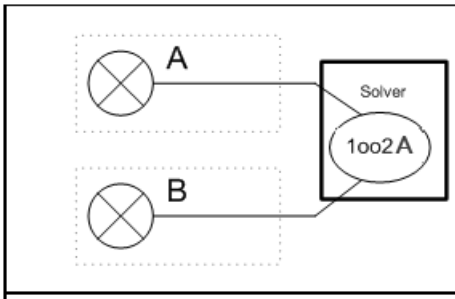
The reconfigurations in A/M configuration are the following:

- 1oo3 -> 1oo2 -> 1oo1
- 2oo3 -> 1oo2 -> 1oo1
- 3oo3 -> 2oo2 -> 1oo1
- MooN -> Moo(N-1)-> Moo(N-2) etc while N-i > M, then M and N are decreased of 1 until 2oo3 configuration
- NooN -> (N-1)oo(N-1) etc until 1oo1

Example: 4oo8 -> 4oo7 -> 4oo6 -> 4oo5 -> 3oo4 -> 2oo3

Following chapters detail S and A configuration for 1oo2 and 2oo3.

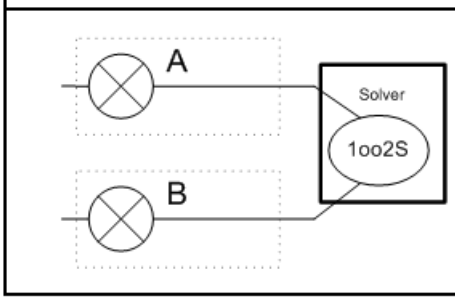
3.3.1. 1oo2S and 1oo2A



Sensor in 1oo2A configuration

Normal: The safety function is inactive
Tripped: The safety function is active

B\A	Normal	At threshold	Invalid
Normal	***Normal***	*Tripped*	***Normal***
At threshold	*Tripped*	*Tripped*	*Tripped*
Invalid	***Normal***	*Tripped*	*Tripped*

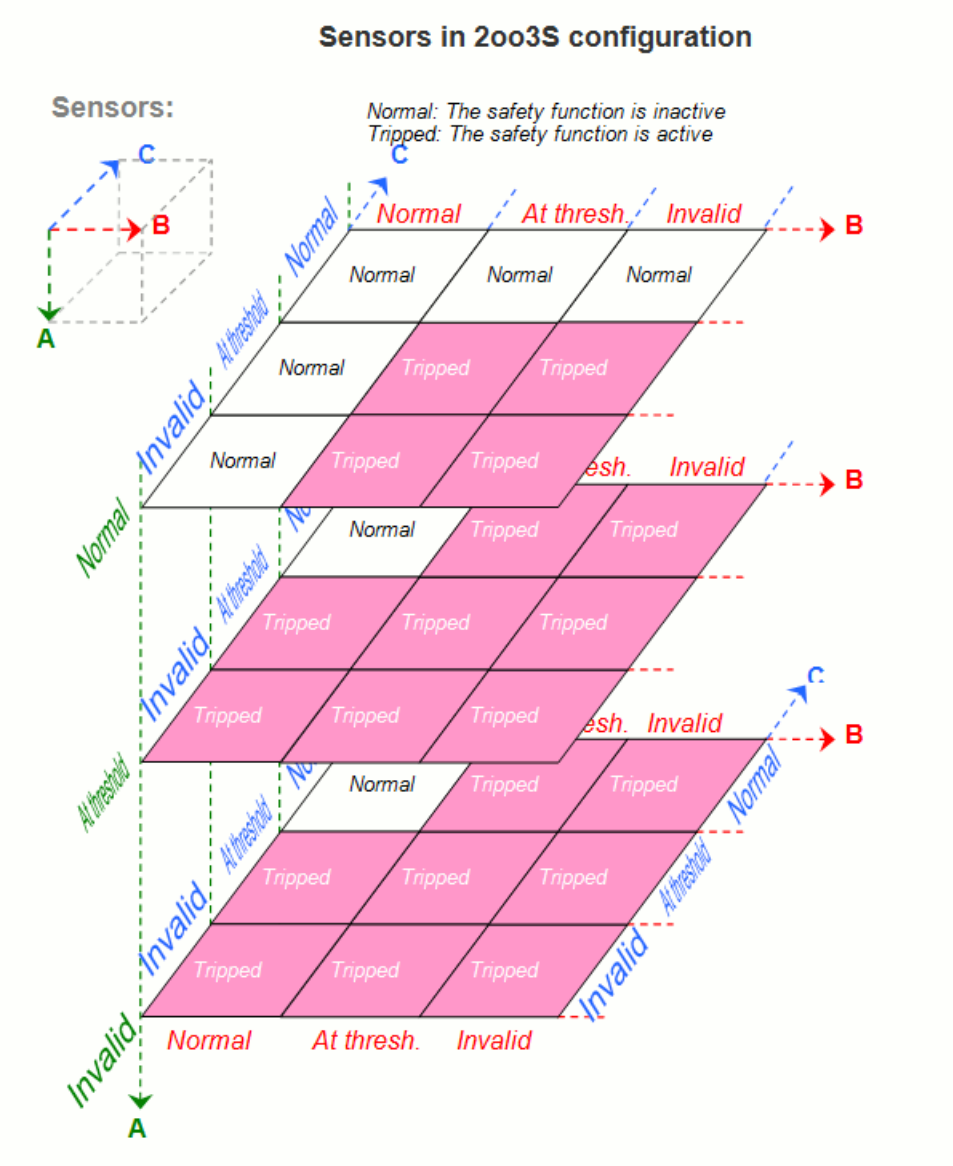


Sensor in 1oo2S configuration

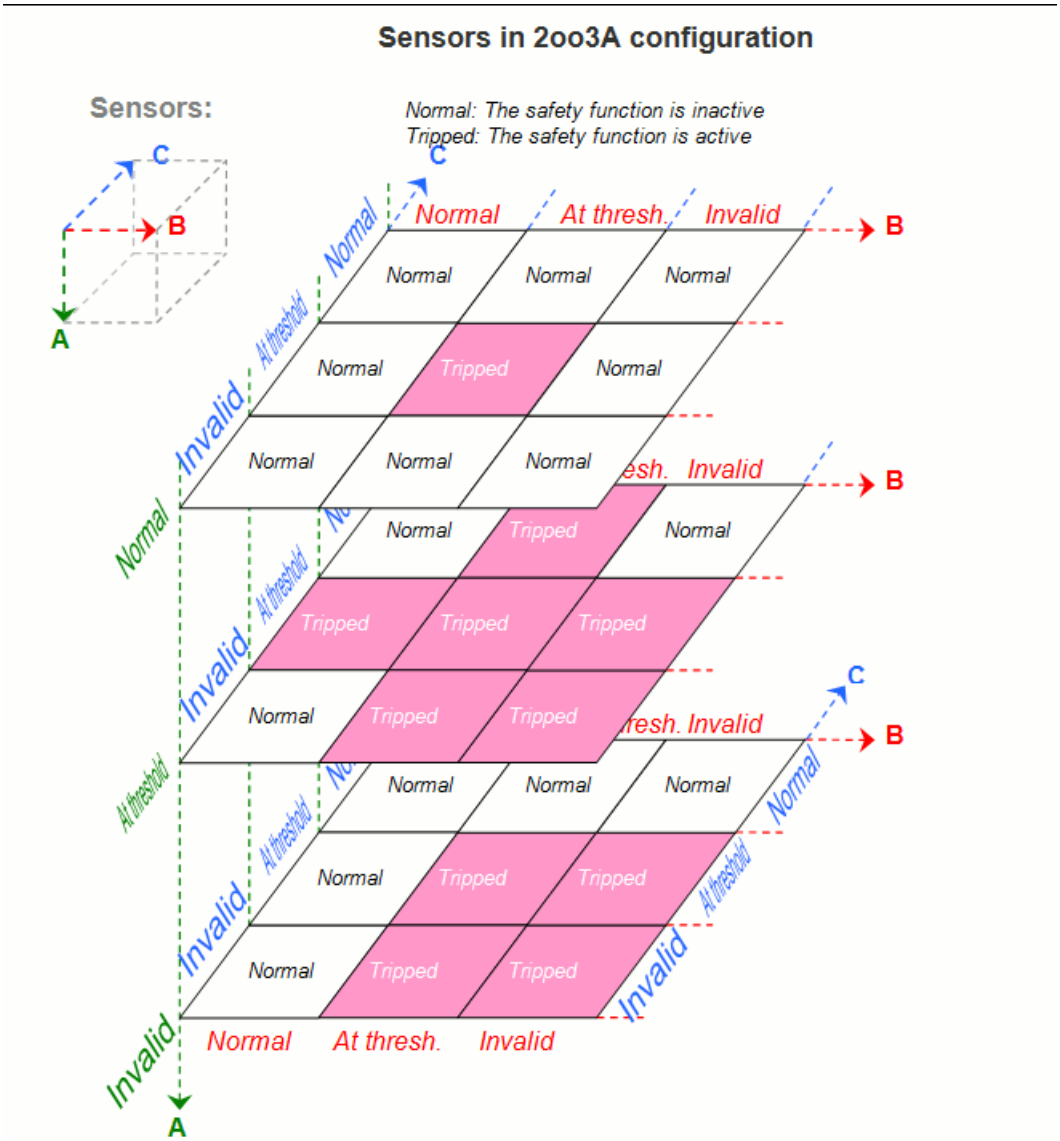
Normal: The safety function is inactive
Tripped: The safety function is active

B \ A	Normal	At threshold	Invalid
Normal	***Normal***	*Tripped*	*Tripped*
At threshold	*Tripped*	*Tripped*	*Tripped*
Invalid	*Tripped*	*Tripped*	*Tripped*

3.3.2. 2oo3S


Sensors in 2oo3S configuration
Sensors:
Normal: The safety function is inactive
Tripped: The safety function is active
A
B
C
Normal
At thresh.
Invalid
Tripped

3.3.3. 2oo3A



3.4. Configuration of channels of a part

You can select a MooN (M out of N) configuration, in this case, the system needs M working sub-systems (out of N) to be available for its safety function. You can also choose a specific configuration. For example, if you need to configure 3 channels as follows: channel1 OR (channel2 AND channel3), select the **manual** button and type: **(1 | (2 & 3))** in the text field. In formula, each channel is replaced by its number. For logical OR, use pipe (|) character. For AND use &. Operators have different priority, you must use parenthesis.

3.5. Take Common Cause Failures into account

You can use Common Cause Failures (CCF) at different levels of architecture. For each level, you can specify a beta-factor. For expert-users, you can display the DDC period configuration (with **Tools**), it let you specify a period (in hour) for CCF test. This period is automatically calculated. Uncheck the period checkbox if and only if you really know what you do.

3.6. Constraints on SIF architecture

The architectural constraints are defined by IEC61508 and 61511 standards to limit the maximum SIL that can be achieved according to the Hardware Fualt Tolerence and caracteristics of components. This mximum SIL

is independent from PFD and PFH. The architectural constraints differ according to the standard. This chapter summarizes the various architectural constraints of a safety instrumented function according to the standard.

3.6.1. Definition

Please refer to standard for more details. Only small extracts are listed in next chapters.

- **SFF :** Safe Falure Fraction
- **HFT :** Hardware Fault Tolerence
- **Field proven/Standard/Non-safety:** The previoux IEC61511 (2003) defined caracteristics (positiv safety/ fild proven/certified etc ...) that was necessary for maximum SIL computation. Theses differences have been removed from computation in 2016.
- **Type A :** Type A component (see IEC 61508)
- **Type B** Type B component (see IEC 61508)

3.6.2. IEC 61508: Route 1H

According to Chapter 7.4.2.2 of IEC 61508, the following table define the maximum reachable SIL depending on the number of hardware faults that are acceptable, and depending on the Safe Failure Fraction.

- **Type A:**

Safe Failure Fraction (SFF)	Hardware fault tolerance (HFT)		
	0	1	2
< 60%	SIL 1	SIL 2	SIL 3
60% - < 90%	SIL 2	SIL 3	SIL 4
90% - < 99%	SIL 3	SIL 4	SIL 4
≥ 99%	SIL 3	SIL 4	SIL 4

- **Type B and others:**

Safe Failure Fraction (SFF)	Hardware fault tolerance (HFT)		
	0	1	2
< 60%	FORBIDDEN	SIL 1	SIL 2
60% - < 90%	SIL 1	SIL 2	SIL 3
90% - < 99%	SIL 2	SIL 3	SIL 4
≥ 99%	SIL 3	SIL 4	SIL 4

3.6.3. IEC 61508: Route 2H

According to Chapter 7.4.2.3 of IEC 61508, the following table define the maximum reachable SIL depending on the hardware faults tolerance is:

SIL	0 HFT	1 HFT	2 HFT
1	✓	✓	✓
2 PFD	✓	✓	✓
2 PFH	⊘	✓	✓
3	⊘	✓	✓
4	⊘	⊘	✓

3.6.4. IEC 61508: Route 2H - Chap 7.4.4.3.2

«For type A elements only, if it is determined that by following the HFT requirements specified in 7.4.4.3.1, for the situation where an HFT greater than 0 is required, it would introduce additional failures and lead to a decrease in the overall safety of the EUC, then a safer alternative architecture with reduced HFT may be implemented. In such a case this shall be justified and documented. The justification shall provide evidence that:

a) compliance with the HFT requirements specified in 7.4.4.3.1 would introduce additional failures and lead to a decrease in the overall safety of the EUC; and

b) if the HFT is reduced to zero, the failure modes, identified in the element performing the safety function, can be excluded because the dangerous failure rate(s) of the identified failure mode(s) are very low compared to the target failure measure for the safety function under consideration (see 7.4.4.1.1 c)). That is, the sum of the dangerous failure frequencies of all serial elements, on which fault exclusion is being claimed, should not exceed 1 % of the target failure measure. Furthermore the applicability of fault exclusions shall be justified considering the potential for systematic faults»

- If HFT ≥ 1 ⇒ SIL 4;
- If HFT = 0 ⇒ SIL 3;

3.6.5. IEC 61511 - Version 2016

Minimum HFT requirements according to SIL

SIL	0 HFT	1 HFT	2 HFT
1	✓	✓	✓
2 PFD	✓	✓	✓
2 PFH	⊘	✓	✓
3	⊘	✓	✓
4	⊘	⊘	✓

3.6.6. IEC 61511 - Version 2016 - Chap 11.4.6

For a SIS or SIS subsystem that does not use FVL or LVL programmable devices and if the minimum HFT as specified in Table 6, would result in additional failures and lead to decreased overall process safety, then the HFT may be reduced. This shall be justified and documented. The justification shall provide evidence that the proposed architecture is suitable for its intended purpose and meets the safety integrity requirements.

- If HFT ≥ 1 ⇒ SIL 4;

- If HFT = 0 ⇒ SIL 3;

3.6.7. How to configure the standard that will be used for contraints

You can choose the standard used to compute the maximum SIL in the **Data and computations**/**Applied standard for contraints** menu. This choice will be applied to every loop of the document, for sensor part and actuator part. If you want to specify a specific standard pour each part of a loop, you can activate the **Data and computations**/ **Configure specific standards in the parts** option. In this case, the architectural tab will contain a **Specific standard in the part** area, which can be used to define a specific standard for each part.

4. Configuration of components

The aim is to specify values for each element of the SIF being studied.

Do this with the tabs of configuration window:

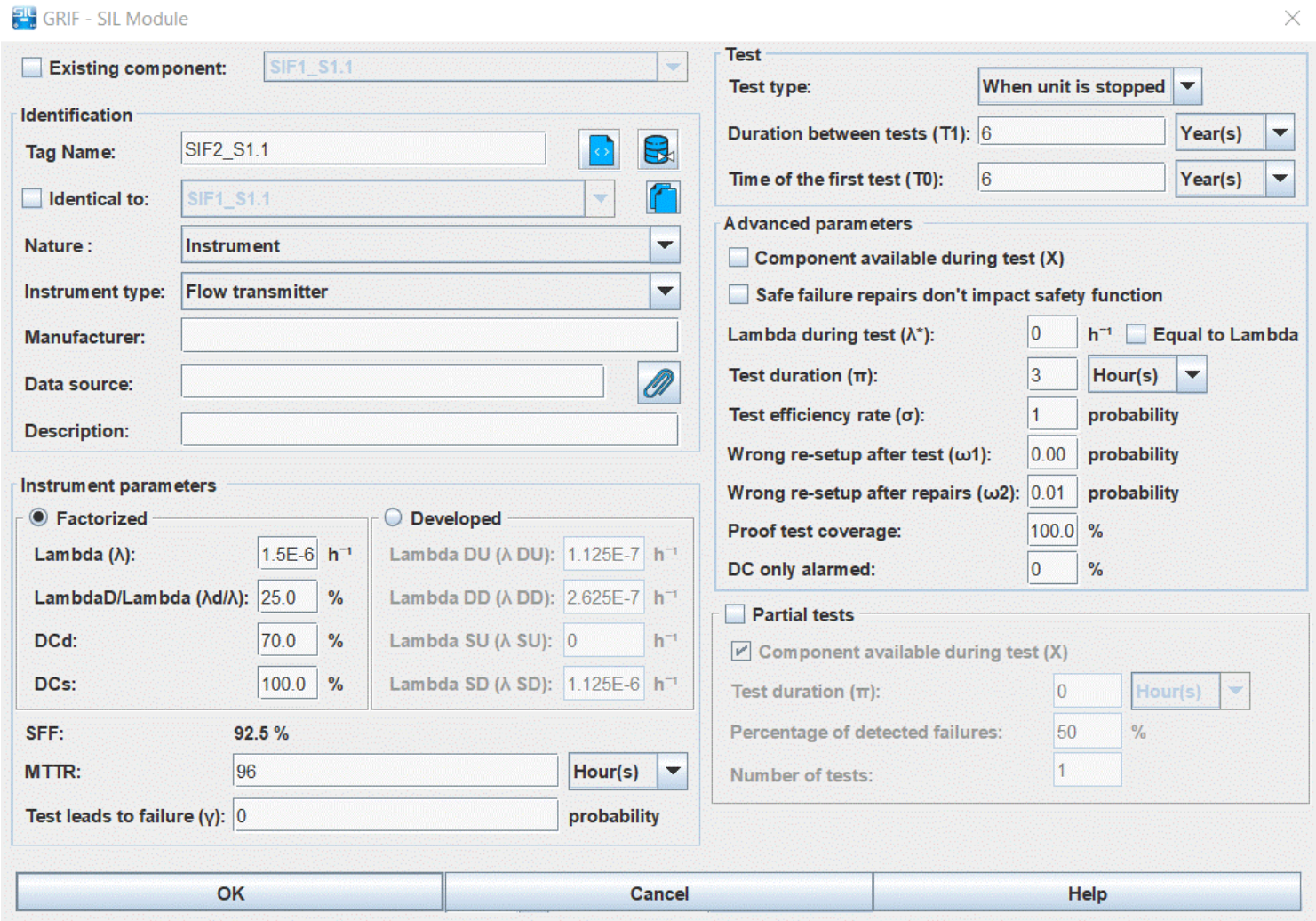
- The **Sensor(s)** tab is used to configure the sensors,
- The **Solver** tab is used to configure the solver,
- The **Actuator(s)** tab is used to configure the actuators
- And the **Parameters** tab contains the definitions of the model's parameters.

In the following chapters, all the numerical values entered can be real numbers, where the decimal separator is a dot. It is possible to write them as such: 0.0000015 or in scientific notation: 1.5E-6

4.1. Configuring the sensors

The sensors of the safety loop can be configured in the **Configuration of components**/**Sensor(s) Part** tab. Each sensor can be accessed separately in the sub-tabs **S1.1**, **S1.2**, etc. The first number (before the dot) is the channel number, the second (after the dot) is the position in the channel.

Configuration of sensors is also accessible by a double click in the input zone on the sensor which user want to set up.



In the following paragraph, "the component" means the sensor.

4.1.1. Existing component

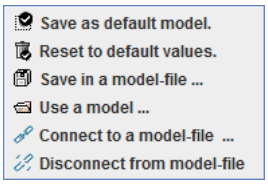
The component may be already used/defined somewhere else in the system. In this case we speak about **existing component**. For example, when a component is in 2 channels. The **existing component** can be selected in a list. It can be a component of the current SIF, of one of another SIF. This options is only available when you have many components of the same type.

4.1.2. Identification

Tag name : component's instrument tag on PID (e.g.: 10 PT 2034 for a sensor, 10 UV 2034 for an actuator, or 10_ESD_06 for solver).

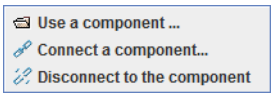
Near to **Tag name** two icons enables you to export or import parameters from a xml file or from a database.

Import/Export components properties in xml format Six actions can be chosen from the drop-down menu, displayed with a left click on the button:



- **Save as default model** Save as default model. : saves the component's characteristics in the default model.
- **Re-initialise to default values** Re-initialyse to default values. : copies into the component the characteristics stored in the default model.
- **Save in a model file** Save in a model-file ... : saves the component's characteristics in a model file, whose location must be specified. This file can be reused or sent to another person.
- **Use a model** Use a model ... : copies into the component the characteristics stored in a model whose location must be specified. The name of the file where the model is located will be displayed at the left of the button.
- **Connect to a model file** Connect to a model-file ... : enables to connect a component to a model of a component stored in a database, whose location must be specified.
- **Disconnect to a model file** Disconnect from model-file : disconnect the component to the model file.

Base of components By clicking on the icon 3 actions are available.



- **Use a component** Use a component ... : copies into the component the characteristics stored in the base of components.
- **Connect a component** Connect a component... : connect the component to a component stored in the base of components. The name of the file where the component is located will be displayed at the left of the button.
- **Disconnect a component** Disconnect to the component : disconnect a component in the base of components.

Identical to : used to specify whether the component is identical to another component of the same type (i.e. a sensor when editing a sensor).

By clicking on the following icon it is possible to **copy another component's parameters**.

This functionality can only be accessed when the SIF have several components of the same type. Only the characteristics **Tag** and **Identical to** are not copied. The components available are the same as those displayed for the functionality **Identical to**.

It is different from **Existing component**. Here the component is not exactly the selected one, they are physically distinct, but they have same parameters. This functionality can only be accessed when the SIF have several components of the same type. If the checkbox is checked, only the **Tag** of the component can be edited (the others are identical to the reference component).

Instrument type : type of instrument used. It is selected from a drop-down menu.

Manufacturer : inform the manufacturer of the component.

Data source : Indicate where reliability data are extracted.

Description : open field where the user can add his own description of the component.

In the base of components, these information are filled in the following columns:

- **ID**
- **REPERE**
- **DESCRIPTION**
- **INSTRUMENTED_TYPE**:All the components type are given in the Appendix C, *List of components*
- **MANUFACTURER**
- **DATA_SOURCE**

4.1.3. Determined character of the component

Determined character of the component : enables you to specify the component's determined character. The component is characterised by one of the three characters available:

- **Non-type A/B** : indicates that the component is operating in negative safety mode (energies to trip) and without self-diagnostic system. Corresponds to the NS type (Non-safety component) of versions previous to 2013.
- **Type B** : indicates that the component is operating in positive safety mode (fail-safe) or equipped with a self-diagnostic system. Corresponds to the S type (Standard component) of versions previous to 2013.
- **Type A** : indicates that the component is operating in positive safety mode (fail-safe) and proven in use (or certified) and equipped with a self-diagnostic system (or implementation of several proof test) and access protected safeguarding the settings of the internal configuration parameters. Corresponds to the F type (Field proven component) of versions previous to 2013.

In the base of components, these information are filled in the **DETERMINED_CHARACTER** column:

- NS for **No-type A/B** component;
- S for **type B** component;
- F for **type A** component;

4.1.4. Test

Test type: enables you to specify the type of test used for the component. Two types of test can be selected from the drop-down menu:

- **Test when unit is stopped**: means that the component is tested when the unit is stopped. The test does not harm the safety function as the unit is no longer working.
- **Test when unit is working**: means that the component is tested when the unit is working. The component is no longer available to carry out its function and this affects the safety function. This can be used when a sensor has been by-passed to be tested and the installation has not been stopped.

it is also possible to specify that the component will undergo no periodic test.

Duration between tests (T1): period of time between two proof tests of the component. The time unit is selected from a drop-down list (**hours**, **days**, **months**, **years**).

Time of the first test (T0): time at which the first test of the component is carried out. The modes for editing this characteristic (value and unit) are the same as for the duration between tests.

In the base of components, these information are filled in the following columns:

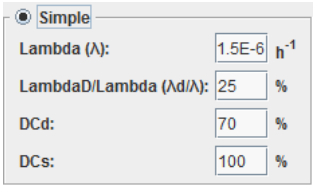
- **TEST_TYPE**:
  - TESTUNITWORK pour **Test unité en marche**;
  - TESTUNITSTOP pour **Test unité à l'arrêt**;
  - EXP pour **Non testé**;
- **T0**: Fist test;
- **T0_UNIT**: HOUR, DAY, MONTH or YEAR;
- **T1**:Duration between tests;
- **T1_UNIT**:: HOUR, DAY, MONTH or YEAR.

4.1.5. Instrument parameters

This part includes all reliability data for a component.

For failure rates 2 different ways can be used to inform them:

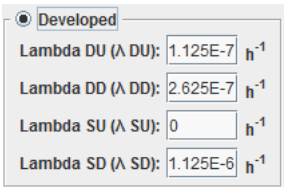
Simple way: with the following parameters:



The value can be edited manually or selected from a drop-down list dusing automatic completion.

- **Lambda** λ: failure rate of the component (h^{-1}).
- **LambdaD/Lambda (λ_d/λ)**: proportion of dangerous failures among the total number of failures.
- **DCd**: on-line diagnostic coverage of dangerous failures and is a rate between 0 and 100%. A 0% rate means that no revealed dangerous failures can be detected.
- **DCs**: on-line diagnostic coverage of safe failures and is a rate between 0 and 100%. A 0% rate means that no revealed safe failures can be detected.

Developped way with the following parameters:



The value can be edited manually or selected from a drop-down list using automatic completion.

- **Lambda DU (λ_{du})**: Dangerous undetected failure rate of the component (h^{-1}).
- **Lambda DD (λ_{dd})**: Dangerous detected failure rate of the component (h^{-1}).

- **Lambda SU (λ_{su})**: Safe undetected failure rate of the component (h^{-1}).
- **Lambda SD (λ_{sd})**: Safe detected failure rate of the component (h^{-1}).

SFF (Safe Failure Fraction): corresponds to the safe failure rate $(\lambda_{sd} + \lambda_{su} + \lambda_{dd}) / \lambda$

It is not an editable field.

MTTR (Mean Time To Repair) in h: mean time between detection of a failure and the repair of the component. The time unit is selected from a drop-down list (**hours** , **days** , **months** , **years**). The value can be edited manually or selected from a drop-down list using automatic completion.

This field is editable only if **DCd** or **λ DD** are not 0 or if **Test type** is equal to **Test when unit is working** .

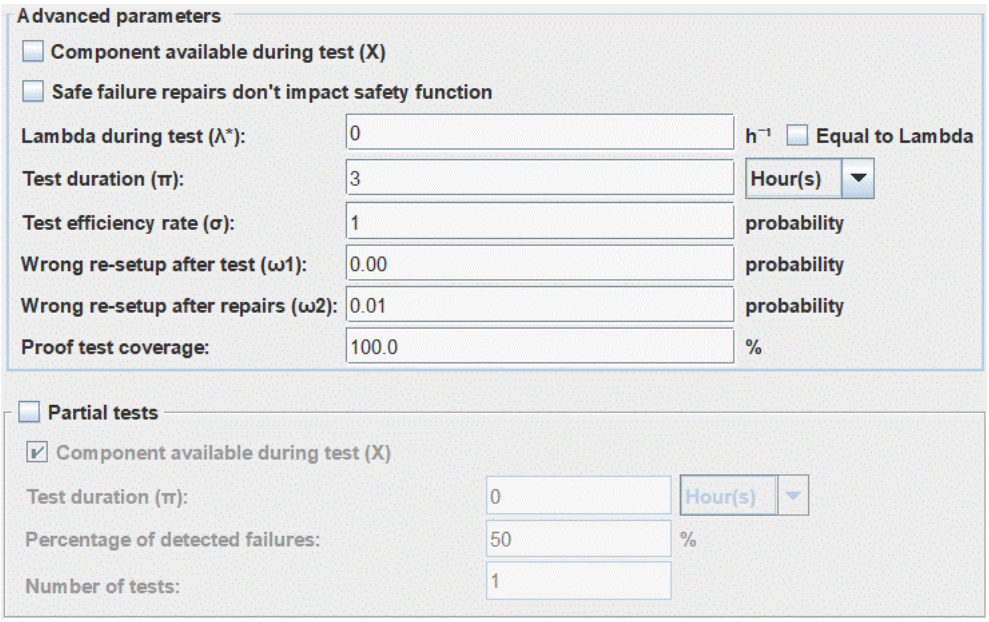
Test leads to failureγ (Gamma): probability [0,1] that the test will cause the hardware to fail. 0 means no test causes any failure, 1 mean every test causes failures. The value can be edited manually or selected from a drop-down list using automatic completion.

This field is editable only if **DCd** or **λ DD** or if **Test type** is different of **Not tested** .

In the base of components, these information are filled in the following columns:

- **MODE** : DEVELOPED or FACTORISED
- **LAMBDA** ;
- **DFF** ;
- **DC_D** ;
- **DC_S** ;
- **LAMBDA_DU** ;
- **LAMBDA_DD** ;
- **LAMBDA_SU** ;
- **LAMBDA_SD** ;
- **MTTR** ;
- **MTTR_UNIT** :HOUR, DAY, MONTH or YEAR;
- **GAMMA** .

The advanced parameters of a sensor can be specified in the **Advanced parameters** part.



The advanced parameters of the sensor are as follows:

- **Component available during test (X)**: specifies whether the component is able to carry out its safety mission during the test (if the checkbox is checked).
- **Lambda during test λ***: failure rate of the component during the test (h^{-1}). The test conditions may cause extra stress and increase the lambda. It is possible to indicate that the value is to be equal to lambda (λ_{du}).
- **Test duration π (Pi)**: period of time necessary for testing the component. The time unit is selected from a drop-down list (**hours**, **days**, **months**, **years**).

This field is editable only if **Test type** is equal to **Test when unit is working**.

- **Test efficiency rate σ (Sigma)** : cover or efficiency rate of the test. The value ranges from 0 (the test never detects anything) to 1(the test always detects the failure).
- **Wrong re-setup after tests**ω1 **(Omega1)**: probability [0,1] of wrong re-setup of the equipment after the test. It is the probability that the component will not be able to carry out its safety mission after being tested by the operator. It can be left at 0 if you consider that the operators and test procedures are infallible (no omission of a by-passed sensor, powering up the motor, etc.).
- **Wrong re-setup after repairs**ω2 **(Omega2)**: probability [0,1] of wrong re-setup of the equipment after the repairs. It is the probability that the component will not be able to carry out its safety mission after being repaired (or changed) by the operator. It can be left at 0 if you consider that the operators and repairs procedures are infallible (powering up the new motor, etc.).
- **Coverage of the proof test** : enables to specify if the component is tested on all of its failures, or if the component is tested only on a part of its failures. If a component is tested on all of its failures, then the coverage of the proof test is 100% (default value). If only a part of the component is tested, then it is possible to specify this coverage by giving a percent of the tested failures.

In the base of components, these information are filled in the following columns:

- **X**: FALSE ou TRUE;
- **LAMBDASTAR**;
- **LAMBDASTAR_EQUALTO_LAMBDA**: FALSE or TRUE;
- **PI**
- **PI_UNIT**:HOUR, DAY, MONTH or YEAR;
- **SIGMA**;
- **OMEGA1**;
- **OMEGA2**.

- **DC only alarmed** :Percentage of detected failure that are only alarmed (non-triggering). This field is available only if channel is in M Mode.

In the base of components, these information are filled in the following column:

- **DC_ONLY_ALARMED** : percent of DC only alarmed.

- **With partial stroking test**: if checked, specifies whether the component takes partial stroking tests into account, as for example the partial stroke testing of a valve gate.
- **Component available during test (X)**: specifies whether the component is able to carry out its safety mission during the test (if the checkbox is checked).
- **Test duration π (Pi)**: period of time necessary for testing the component. The time unit is selected from a drop-down list (**hours**, **days**, **months**, **years**).
- **Proportion of detected failure** :proportion of hidden failures detected during partial stroking tests (0-100%). 0% means no failure is detected, 100% means every failure is detected. LThe value can be edited manually or selected from a drop-down list using automatic completion.
- **Number of tests**: number of partial stroking tests carried out between two full tests.

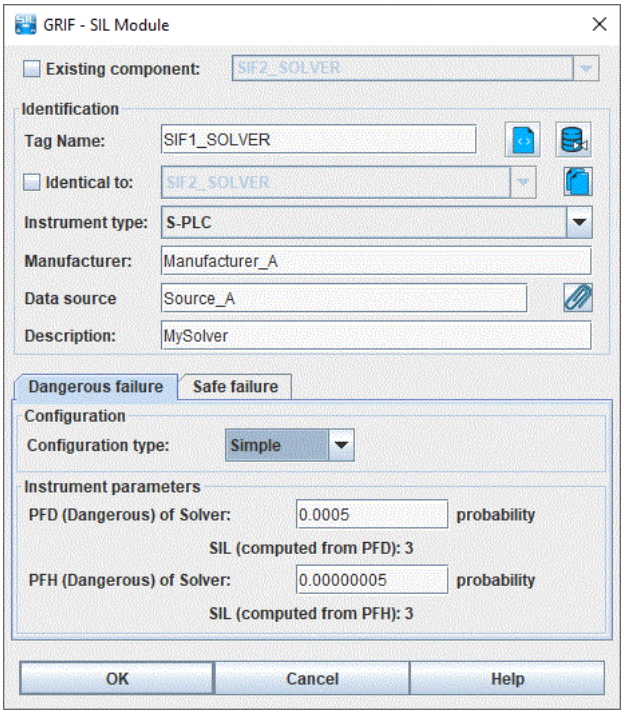
In the base of components, these information are filled in the following columns:

- **WITH_PARTIALSTROKING**: TRUE ou FALSE;

- **PARTIALSTROKING_X**: TRUE ou FALSE;
- **PARTIALSTROKING_PI**;
- **PARTIALSTROKING_PI_UNIT**:HOUR, DAY, MONTH ou YEAR;
- **PARTIALSTROKING_EFFICACITY**;
- **PARTIALSTROKING_NBTESTS**.

4.2. Configuring the solver

The solver of the safety loop can be configured in the **Solver** tab.



In the following paragraph, "the component" means the sensor.

4.2.1. Solveur existant

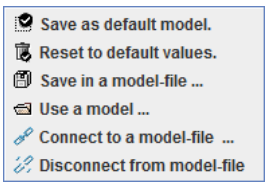
The solver may be used in many SIF, and may has been defined in an **existing solver** of the document. The reference-solver can be selected in a list. This options is only available when you have many SIF.

4.2.2. Identification

Tag name : component's instrument tag on PID (e.g.: 10 PT 2034 for a sensor, 10 UV 2034 for an actuator, or 10_ESD_06 for solver).

Near to **Tag name** two icons enables you to export or import parameters from a xml file or from a database.

Import/Export components properties in xml format Six actions can be chosen from the drop-down menu, displayed with a left click on the button:



- **Save as default model** Save as default model. : saves the component's characteristics in the default model.
- **Re-initialise to default values** Re-initialyse to default values. : copies into the component the characteristics stored in the default model.
- **Save in a model file** Save in a model-file ... : saves the component's characteristics in a model file, whose location must be specified. This file can be reused or sent to another person.
- **Use a model** Use a model ... : copies into the component the characteristics stored in a model whose location must be specified. The name of the file where the model is located will be displayed at the left of the button.
- **Connect to a model file** Connect to a model-file ... : enables to connect a component to a model of a component stored in a database, whose location must be specified.
- **Disconnect to a model file** Disconnect from model-file : disconnect the component to the model file.

Base of components By clicking on the icon 3 actions are available.

Use a component ...
Connect a component...
Disconnect to the component

- **Use a component** Use a component ... : copies into the component the characteristics stored in the base of components.
- **Connect a component** Connect a component... : connect the component to a component stored in the base of components. The name of the file where the component is located will be displayed at the left of the button.
- **Disconnect a component** Disconnect to the component : disconnect a component in the base of components.

Identical to : used to specify whether the component is identical to another component of the same type (i.e. a sensor when editing a sensor).

By clicking on the following icon it is possible to **copy another component's parameters**.

This functionality can only be accessed when the SIF have several components of the same type. Only the characteristics **Tag** and **Identical to** are not copied. The components available are the same as those displayed for the functionality **Identical to**.

It is different from **Existing component**. Here the component is not exactly the selected one, they are physically distinct, but they have same parameters. This functionality can only be accessed when the SIF have several components of the same type. If the checkbox is checked, only the **Tag** of the component can be edited (the others are identical to the reference component).

Instrument type : type of instrument used. It is selected from a drop-down menu.

Manufacturer : inform the manufacturer of the component.

Data source : Indicate where reliability data are extracted.

Description : open field where the user can add his own description of the component.

In the base of components, these information are filled in the following columns:

- **ID**
- **REPERE**
- **DESCRIPTION**
- **INSTRUMENTED_TYPE**:All the components type are given in the Appendix C, *List of components*
- **MANUFACTURER**
- **DATA_SOURCE**

4.2.3. Configuration

There is two types of failures for the solver: **Dangerous failure** and **Safe failure**.

Configuration type : specifies the solver's configuration type of the selected failure. Two types of configuration can be selected from the drop-down menu:

- **Simple** : the solver is modelled by a constant law.
- **Advanced**: the solver is modelled by a full periodic test law.
- **Specific law** :user can be choose the law used to modelize the solver.

In the base of components, these information are filled in the column **TEST_TYPE**

- **CST** : the solver is modelled by a constant law;
- **TPC** : the solver is modelled by a full periodic test law;
- **LAW_SPEC** : :user can be choose the law used to modelize the solver;

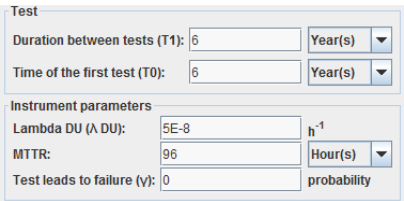
4.2.4. Instrument parameters

The parameters of the solver are described under **Instrument parameters**. They depend on the type of configuration which has been selected.

In the case of a simple configuration, the parameters are as follows:

- **PFD of solver**: probability that the solver will not work when triggered. This value can be edited manually or selected from a drop-down list displaying all of the parameters with a **Probability** dimension.
- **SIL (computed from PFD)**: automatically displays the solver's SIL computed based on the solver's PFD.
- **PFH of solver**: the PFH of the solver, given by manufacturer or by experience feedbacks.
- **SIL (computed from PFH)**: automatically displays the solver's SIL computed based on the solver's PFH.

In the case of an advanced configuration, the parameters are as follows:



- **Duration between tests (T1)**: period of time between two proof tests of the component. The time unit is selected from a drop-down list (**hours**, **days**, **months**, **years**). This value can be edited manually or using automatic completion.
- **Time of the first test (T0)**: time at which the first test of the component is carried out. The modes for editing this characteristic (value and unit) are the same as for the duration between tests.

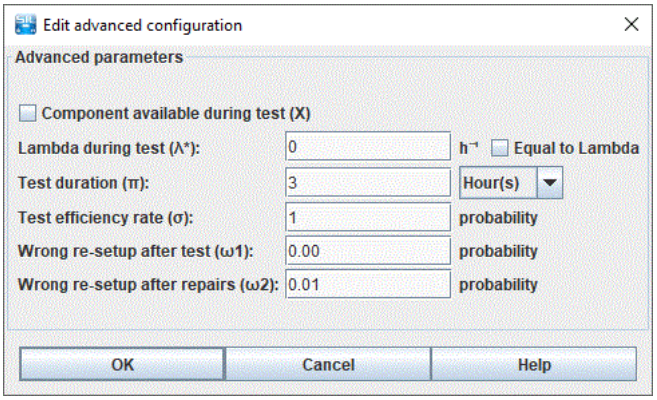
- **Lambda DU (λ_{du})**: Dangerous undetected failure rate of the component (h^{-1}). This value can be edited manually or using automatic completion.
- **MTTR (Mean Time To Repair) in h**: mean time between detection of a failure and the repair of the component. The time unit is selected from a drop-down list (**hours**, **days**, **months**, **years**). This value can be edited manually or using automatic completion.
- **Test leads to failureγ (Gamma)**: probability [0,1] that the test will cause the hardware to fail. 0 means no test causes any failure, 1 mean every test causes failures. This value can be edited manually or using automatic completion.

In the base of components, these information are filled in the following columns:

- **PFD**
- **PFH**
- **T0**: First test;

- **T0_UNIT**:HOUR, DAY, MONTH or YEAR;
- **T1**: Intervalle between test;
- **T1_UNIT**:HOUR, DAY, MONTH or YEAR.
- **LAMBDA_DU** ;
- **MTTR**;
- **MTTR_UNIT**: HOUR, DAY, MONTH or YEAR;
- **GAMMA**.

Other parameters can be accessed by left clicking on the **Advanced configuration ...** button (only for a solver configured in advanced mode).



- **Component available during test (X)**: specifies whether the component is able to carry out its safety mission during the test (if the checkbox is checked).
- **Lambda during test λ***: failure rate of the component during the test (h^{-1}). The test conditions may cause extra stress and increase the lambda. It is possible to indicate that the value is to be equal to lambda (λ_{du}).
- **Test duration π (Pi)**: period of time necessary for testing the component. The time unit is selected from a drop-down list (**hours**, **days**, **months**, **years**).

This field is editable only if **Test type** is equal to **Test when unit is working**.

- **Test efficiency rate σ (Sigma)** : cover or efficiency rate of the test. The value ranges from 0 (the test never detects anything) to 1(the test always detects the failure).
- **Wrong re-setup after tests**ω1 **(Omega1)**: probability [0,1] of wrong re-setup of the equipment after the test. It is the probability that the component will not be able to carry out its safety mission after being tested by the operator. It can be left at 0 if you consider that the operators and test procedures are infallible (no omission of a by-passed sensor, powering up the motor, etc.).
- **Wrong re-setup after repairs**ω2 **(Omega2)**: probability [0,1] of wrong re-setup of the equipment after the repairs. It is the probability that the component will not be able to carry out its safety mission after being repaired (or changed) by the operator. It can be left at 0 if you consider that the operators and repairs procedures are infallible (powering up the new motor, etc.).
- **Coverage of the proof test** : enables to specify if the component is tested on all of its failures, or if the component is tested only on a part of its failures. If a component is tested on all of its failures, then the coverage of the proof test is 100% (default value). If only a part of the component is tested, then it is possible to specify this coverage by giving a percent of the tested failures.

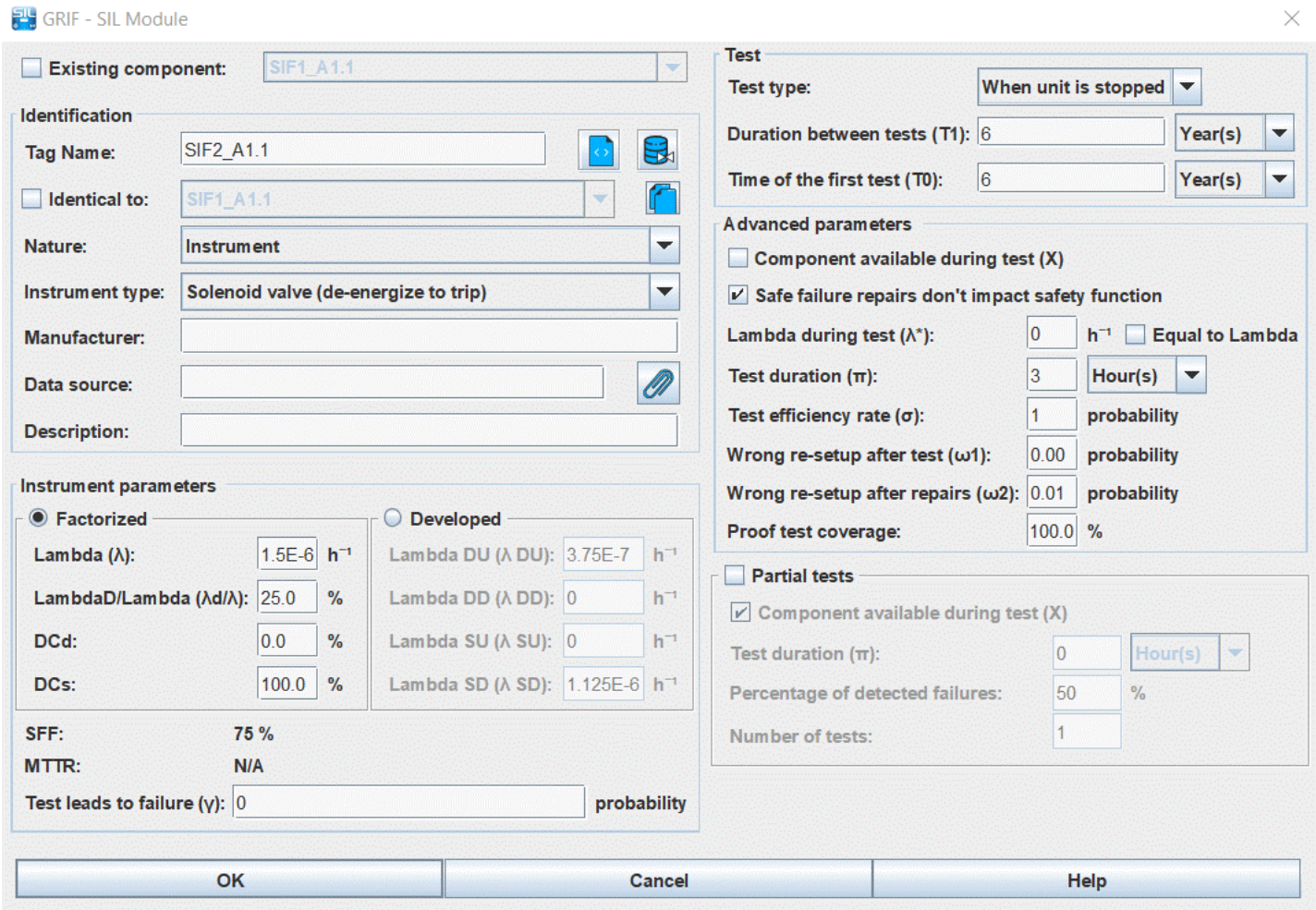
In case of specific law, user can be choose in all law implemented in Albizia (cf. Appendix E, *Law*)

4.3. Configuring the actuators

The actuators of the safety loop can be configured in the **Configuration of componentsActuator(s) Parts** tab. The actuators can be classified as follows:

- **Main actuators**: they have 0, 1 or 2 sub-actuators.
- **Sub-actuators**: they are set up in series with their respective actuators. The sub-actuators of a same main actuator are set up in series (2oo2) or parallel (1oo2).

Each main actuator can be accessed separately in the sub-tabs **A1.1**, **A1.2**, ..., and each sub-actuator in the sub-tabs **A1.1a**, **A1.1b**, ...



In the following paragraph, the actuator (main or sub) will be called "the component".

4.3.1. Existing component

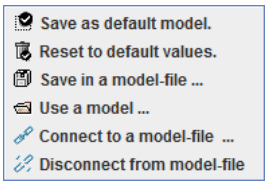
The component may be already used/defined somewhere else in the system. In this case we speak about **existing component**. For example, when a component is in 2 channels. The **existing component** can be selected in a list. It can be a component of the current SIF, of one of another SIF. This options is only available when you have many components of the same type.

4.3.2. Identification

Tag name : component's instrument tag on PID (e.g.: 10 PT 2034 for a sensor, 10 UV 2034 for an actuator, or 10_ESD_06 for solver).

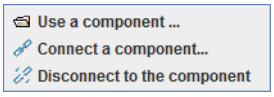
Near to **Tag name** two icons enables you to export or import parameters from a xml file or from a database.

Import/Export components properties in xml format Six actions can be chosen from the drop-down menu, displayed with a left click on the button:



- **Save as default model** Save as default model. : saves the component's characteristics in the default model.
- **Re-initialise to default values** Re-initialyse to default values. : copies into the component the characteristics stored in the default model.
- **Save in a model file** Save in a model-file ... : saves the component's characteristics in a model file, whose location must be specified. This file can be reused or sent to another person.
- **Use a model** Use a model ... : copies into the component the characteristics stored in a model whose location must be specified. The name of the file where the model is located will be displayed at the left of the button.
- **Connect to a model file** Connect to a model-file ... : enables to connect a component to a model of a component stored in a database, whose location must be specified.
- **Disconnect to a model file** Disconnect from model-file : disconnect the component to the model file.

Base of components By clicking on the icon 3 actions are available.



- **Use a component** Use a component ... : copies into the component the characteristics stored in the base of components.
- **Connect a component** Connect a component... : connect the component to a component stored in the base of components. The name of the file where the component is located will be displayed at the left of the button.
- **Disconnect a component** Disconnect to the component : disconnect a component in the base of components.

Identical to : used to specify whether the component is identical to another component of the same type (i.e. a sensor when editing a sensor).

By clicking on the following icon it is possible to **copy another component's parameters**.

This functionality can only be accessed when the SIF have several components of the same type. Only the characteristics **Tag** and **Identical to** are not copied. The components available are the same as those displayed for the functionality **Identical to**.

It is different from **Existing component**. Here the component is not exactly the selected one, they are physically distinct, but they have same parameters. This functionality can only be accessed when the SIF have several components of the same type. If the checkbox is checked, only the **Tag** of the component can be edited (the others are identical to the reference component).

Instrument type : type of instrument used. It is selected from a drop-down menu.

Manufacturer : inform the manufacturer of the component.

Data source : Indicate where reliability data are extracted.

Description : open field where the user can add his own description of the component.

In the base of components, these information are filled in the following columns:

- **ID**
- **REPERE**
- **DESCRIPTION**
- **INSTRUMENTED_TYPE**:All the components type are given in the Appendix C, *List of components*
- **MANUFACTURER**

- **DATA_SOURCE**

4.3.3. Determined character of the component

Determined character of the component : enables you to specify the component's determined character. The component is characterised by one of the three characters available:

- **Non-type A/B** : indicates that the component is operating in negative safety mode (energies to trip) and without self-diagnostic system. Corresponds to the NS type (Non-safety component) of versions previous to 2013.
- **Type B** : indicates that the component is operating in positive safety mode (fail-safe) or equipped with a self-diagnostic system. Corresponds to the S type (Standard component) of versions previous to 2013.
- **Type A** : indicates that the component is operating in positive safety mode (fail-safe) and proven in use (or certified) and equipped with a self-diagnostic system (or implementation of several proof test) and access protected safeguarding the settings of the internal configuration parameters. Corresponds to the F type (Field proven component) of versions previous to 2013.

In the base of components, these information are filled in the **DETERMINED_CHARACTER** column:

- NS for **No-type A/B** component;
- S for **type B** component;
- F for **type A** component;

4.3.4. Test

Test type: enables you to specify the type of test used for the component. Two types of test can be selected from the drop-down menu:

- **Test when unit is stopped**: means that the component is tested when the unit is stopped. The test does not harm the safety function as the unit is no longer working.
- **Test when unit is working**: means that the component is tested when the unit is working. The component is no longer available to carry out its function and this affects the safety function. This can be used when a sensor has been by-passed to be tested and the installation has not been stopped.

it is also possible to specify that the component will undergo no periodic test.

Duration between tests (T1): period of time between two proof tests of the component. The time unit is selected from a drop-down list (**hours**, **days**, **months**, **years**).

Time of the first test (T0): time at which the first test of the component is carried out. The modes for editing this characteristic (value and unit) are the same as for the duration between tests.

In the base of components, these information are filled in the following columns:

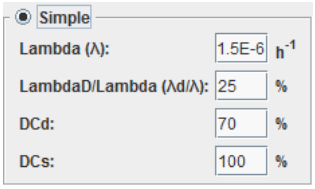
- **TEST_TYPE**:
  - TESTUNITWORK pour **Test unité en marche**;
  - TESTUNITSTOP pour **Test unité à l'arrêt**;
  - EXP pour **Non testé**;
- **T0**: Fist test;
- **T0_UNIT**: HOUR, DAY, MONTH or YEAR;
- **T1**:Duration between tests;
- **T1_UNIT**:: HOUR, DAY, MONTH or YEAR.

4.3.5. Instrument parameters

This part includes all reliability data for a component.

For failure rates 2 different ways can be used to inform them:

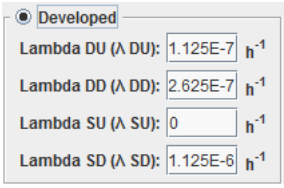
Simple way: with the following parameters:



The value can be edited manually or selected from a drop-down list dusing automatic completion.

- **Lambda** λ: failure rate of the component (h^{-1}).
- **LambdaD/Lambda (λ_d/λ)**: proportion of dangerous failures among the total number of failures.
- **DCd**: on-line diagnostic coverage of dangerous failures and is a rate between 0 and 100%. A 0% rate means that no revealed dangerous failures can be detected.
- **DCs**: on-line diagnostic coverage of safe failures and is a rate between 0 and 100%. A 0% rate means that no revealed safe failures can be detected.

Developped way with the following parameters:



The value can be edited manually or selected from a drop-down list using automatic completion.

- **Lambda DU (λ_{du})**: Dangerous undetected failure rate of the component (h^{-1}).
- **Lambda DD (λ_{dd})**: Dangerous detected failure rate of the component (h^{-1}).
- **Lambda SU (λ_{su})**: Safe undetected failure rate of the component (h^{-1}).
- **Lambda SD (λ_{sd})**: Safe detected failure rate of the component (h^{-1}).

SFF (Safe Failure Fraction): corresponds to the safe failure rate $(\lambda_{sd} + \lambda_{su} + \lambda_{dd}) / \lambda$

It is not an editable field.

MTTR (Mean Time To Repair) in h: mean time between detection of a failure and the repair of the component. The time unit is selected from a drop-down list (**hours** , **days** , **months** , **years**). The value can be edited manually or selected from a drop-down list using automatic completion.

This field is editable only if **DCd** or **λ DD** are not 0 or if **Test type** is equal to **Test when unit is working** .

Test leads to failure γ (Gamma): probability [0,1] that the test will cause the hardware to fail. 0 means no test causes any failure, 1 mean every test causes failures. The value can be edited manually or selected from a drop-down list using automatic completion.

This field is editable only if **DCd** or **λ DD** or if **Test type** is different of **Not tested** .

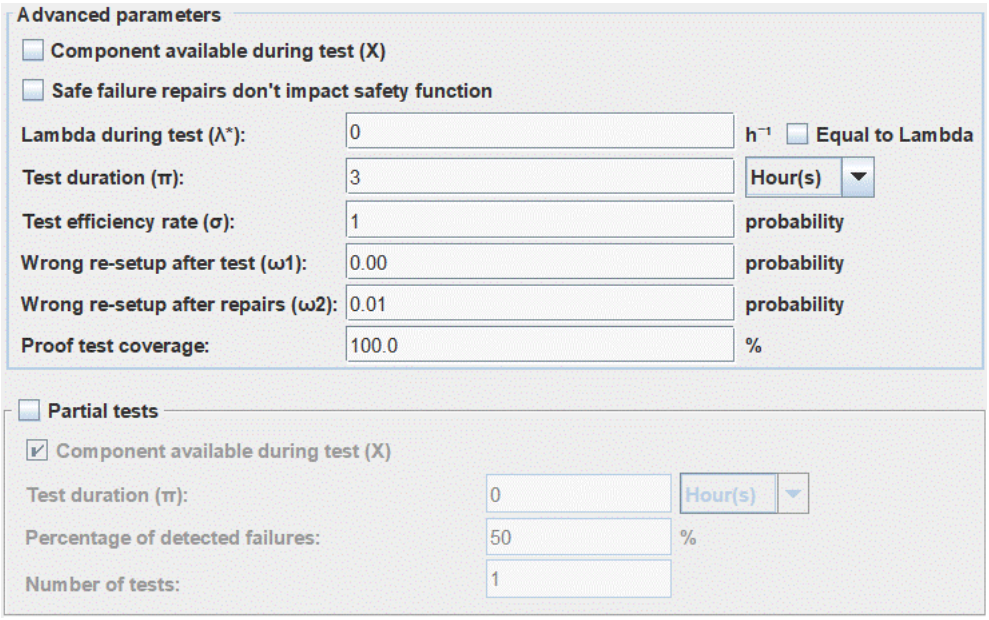
In the base of components, these information are filled in the following columns:

- **MODE** : DEVELOPED or FACTORISED
- **LAMBDA** ;

- **DFF** ;
- **DC_D** ;
- **DC_S** ;
- **LAMBDA_DU** ;
- **LAMBDA_DD** ;
- **LAMBDA_SU** ;
- **LAMBDA_SD** ;
- **MTTR** ;
- **MTTR_UNIT** :HOUR, DAY, MONTH or YEAR;
- **GAMMA** .

Sub-actuators do not have a **Determined character of the component** characteristic. The section relating to this characteristic does therefore not appear when configuring the sub-actuators. As a rule, the sub-actuator has the same character as the one defined for its main actuator.

The advanced parameters of an actuator (main or sub) can be specified in the **Advanced parameters** part.



The advanced parameters of the actuator are as follows:

- **Safe failure repairs don't impact safety function** : if the case is checked, during safe failure repairs have no impact on safety function.
- **Component available during test (X)**: specifies whether the component is able to carry out its safety mission during the test (if the checkbox is checked).
- **Lambda during test λ***: failure rate of the component during the test (h^{-1}). The test conditions may cause extra stress and increase the lambda. It is possible to indicate that the value is to be equal to lambda (λ_{du}).
- **Test duration π (Pi)**: period of time necessary for testing the component. The time unit is selected from a drop-down list (**hours**, **days**, **months**, **years**).

  This field is editable only if **Test type** is equal to **Test when unit is working**.

- **Test efficiency rate σ (Sigma)** : cover or efficiency rate of the test. The value ranges from 0 (the test never detects anything) to 1(the test always detects the failure).
- **Wrong re-setup after tests**ω1 **(Omega1)**: probability [0,1] of wrong re-setup of the equipment after the test. It is the probability that the component will not be able to carry out its safety mission after being tested by the operator. It can be left at 0 if you consider that the operators and test procedures are infallible (no omission of a by-passed sensor, powering up the motor, etc.).
- **Wrong re-setup after repairs**ω2 **(Omega2)**: probability [0,1] of wrong re-setup of the equipment after the repairs. It is the probability that the component will not be able to carry out its safety mission after being repaired

(or changed) by the operator. It can be left at 0 if you consider that the operators and repairs procedures are infallible (powering up the new motor, etc.).

- **Coverage of the proof test** : enables to specify if the component is tested on all of its failures, or if the component is tested only on a part of its failures. If a component is tested on all of its failures, then the coverage of the proof test is 100% (default value). If only a part of the component is tested, then it is possible to specify this coverage by giving a percent of the tested failures.

In the base of components, these information are filled in the following columns:

- **X**: FALSE ou TRUE;
- **LAMBDASTAR**;
- **LAMBDASTAR_EQUALTO_LAMBDA**: FALSE or TRUE;
- **PI**
- **PI_UNIT**:HOUR, DAY, MONTH or YEAR;
- **SIGMA**;
- **OMEGA1**;
- **OMEGA2**.

- **With partial stroking test**: if checked, specifies whether the component takes partial stroking tests into account, as for example the partial stroke testing of a valve gate.
- **Component available during test (X)**: specifies whether the component is able to carry out its safety mission during the test (if the checkbox is checked).
- **Test duration π (Pi)**: period of time necessary for testing the component. The time unit is selected from a drop-down list (**hours**, **days**, **months**, **years**).
- **Proportion of detected failure** :proportion of hidden failures detected during partial stroking tests (0-100%). 0% means no failure is detected, 100% means every failure is detected. LThe value can be edited manually or selected from a drop-down list using automatic completion.
- **Number of tests**: number of partial stroking tests carried out between two full tests.

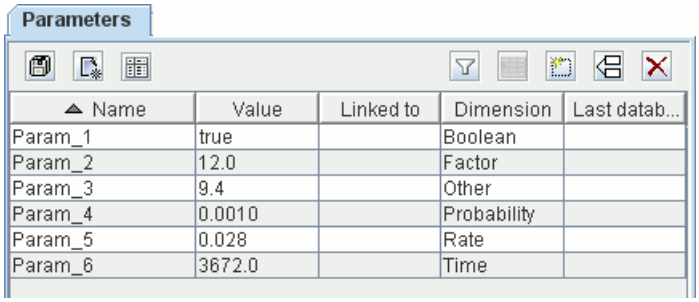
Sub-actuators do not take partial stroking tests into account. There is therefore no Partial stroking test section in the sub-actuator configuration tab.

In the base of components, these information are filled in the following columns:

- **WITH_PARTIALSTROKING**: TRUE ou FALSE;
- **PARTIALSTROKING_X**: TRUE ou FALSE;
- **PARTIALSTROKING_PI**;
- **PARTIALSTROKING_PI_UNIT**:HOUR, DAY, MONTH ou YEAR;
- **PARTIALSTROKING_EFFICACITY**;
- **PARTIALSTROKING_NBTESTS**.

4.4. Editing the parameters

The parameters table contains all of the model's parameters.



Parameters

Name	Value	Linked to	Dimension	Last datab...
Param_1	true		Boolean	
Param_2	12.0		Factor	
Param_3	9.4		Other	
Param_4	0.0010		Probability	
Param_5	0.028		Rate	
Param_6	3672.0		Time	

Each parameter is defined by the following:

- **Name**: the parameter's name (unique).
- **Value**: the parameter's value (consistent with its dimension).
- **Linked to**: the identifier of the database to which the parameter is linked.

- **Dimension**: the parameter's dimension. It is selected from a drop-down menu: Boolean, factor, probability, rate, time, other.
- **Last database**: last database used to update the parameter.

The parameters can then be used to fill in the characteristics of the safety loop's components.

This table can be accessed using the tabs situated on the right but can also be opened in another window via the menu **Data and Computations - Edit parameters**.

4.5. Edition of data table

Data table is used to have a global view of all components. There are 3 different tables:

- **Table of components**:with all SIF components. It is made up of 4 columns i.e. code, tag name, description and type.
- **Edit sensors** gathers in a same table all information for sensors (code, tag name, description, law, law parameters, etc...).
- **Edit actuators** gathers in a same table all information for actuators (code, tag name, description, law, law parameters, etc...).
- **Edit solvers** gathers in a same table all information for solvers (code, tag name, description, law, lawparameters, etc...)
- **Edit SIF** gathers in a same table all information for SIF (code, description, SIL/RRF values obtaned or required, etc...)

Date table is the only place where you can fix the behavior of a component:

- **Degraded Operation Analysis** : it is possible to force the component behavior.
  - By default: the component failure follows the parameters indicated by the user;
  - Dangerous detected failure forced: equipment is always in dangerous detected failure;
  - Dangerous undetected failure forced: equipment is always in dangerous undetected failure.

In the base of components, these information are filled in the following column:

- **BEHAVIOR** : DEFAULT, FAILURE_DD or FAILURE_DU;

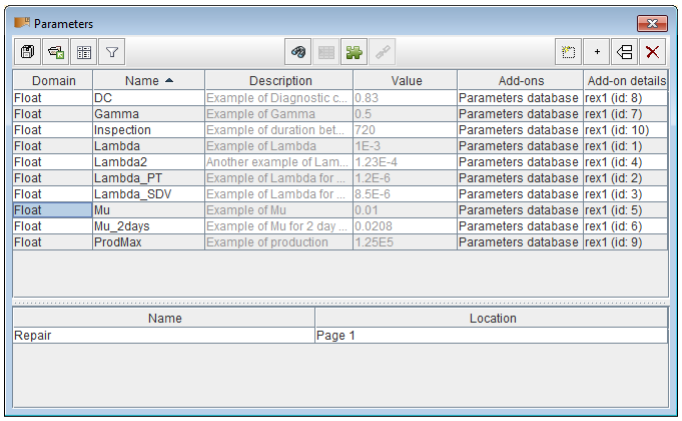
It is possible to display these tables. For that,refer you to **Section 1, “Description of the Tables”**

5. The parameters

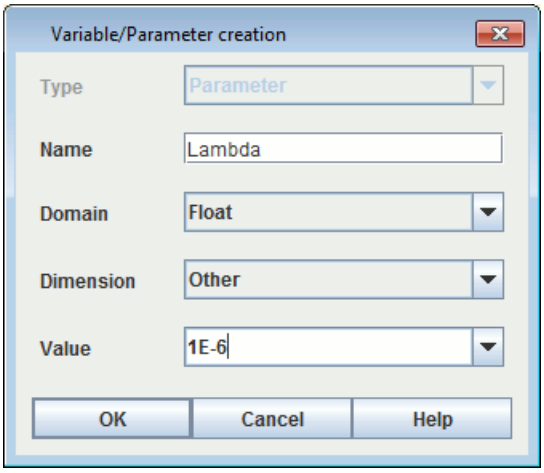
It is possible to create constants which can be booleans, integers or reals. These parameters can then be used for the configuration of different elements of the model (laws, events, transitions, ...)

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.

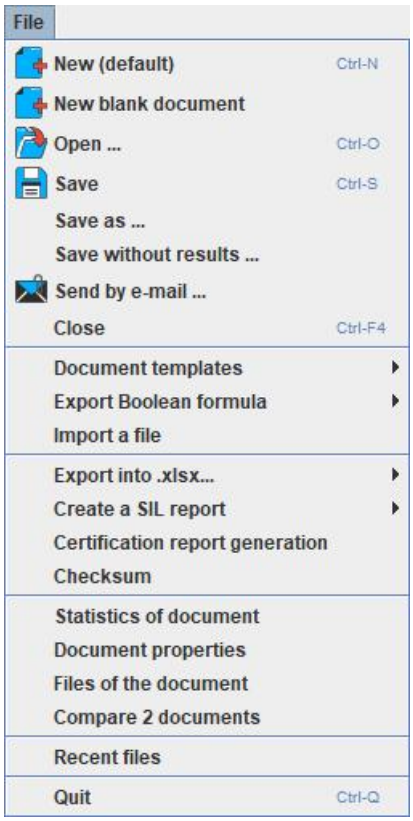
Others additional fields are available in the parameters' table.

Unit		enables to define an unit of the parameter
Uncertainties	**Activate uncertainty**	enables to define the parameter as an uncertainty law
	Law	enables to define the uncertainty law. The law is editable and taken into account only if **Activate uncertainty** is selected in the parameter. The uncertainties laws are detailed here Section 2, “Uncertainties on the parameters”
	Macro	if the parameter is defined by an uncertainty law, and if two events use this same parameter, then the user can choose to use the same uncertainty value for the two events, (**Macro** unselected) or values distinctly computed (**Macro** selected).
Add-On		enables to define the parameter by a GRIF add-on SIL is delivered by default with 2 add-ons 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 8.1, “Database of parameters”. **Beta (61508)** : is an add-on which enables the user to calculate the value of his parameter (β) from a set of questions defined by the IEC 61508-6 Table D.1 standard - for the captors and finals elements.
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.
Beta (61508)	**MooN**	Define the configuration of the system (in functional logic) to use to calculate the beta.
	Beta	Displays a button allowing to modify the choices made in the Table D.1 of the standard IEC 61508-6

6. Menus presentation

6.1. 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 18, “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/SIL

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 **Save without results ...** action saves the document without its result bank that can be be very heavy. It will help users to send their file by e-mail.

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 1.2, “Executables”

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

The menu **Document templates** includes features related to document reuse and pre-configuration, see Section 18, “Document template”.

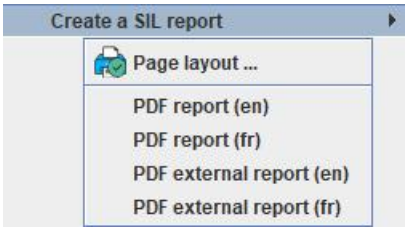
The **Export Boolean formula** menu contains several export actions that can be used to generate a Boolean formula that can be opened with a Fault-Tree software.

The **Export .dag (selected elements)** action exports into .dag the selected elements.
The **Export .dag (ALBIZIA)** action export into ALBIZIA format (containing CCF declatation).
The **Export .xml Open PSA** action exports into OpenPSA file format.

Function **Import a file** allows you to import another document into the current document. A new page will be created from the name of the imported file.

The **Export into .xlsx...** action save all SIF in an Excel file. Each SIF is described in 2 tabs, one for description and result synthesis, the other one for SIF configuration.

Menu **Create a SIL report** allows you to configure and generate a complete PDF report, in French or English, of the instrumented safety loops present in the document. For more information on configuring PDF reports of SIL, refer to Report.



The **Certification report generation** action computes every file in a directory and generates a synthesis of the results in an .xlsx file.

The **Checksum** menu is used to generate the checksum of the safety loop, or to verify if a safety loop has the same checksum of a given one.

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 16, “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 17, “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 13, “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.

6.2. 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** can perform these actions on curves or comments. It is also possible to duplicate a loop by right-clicking on it and using the contextual action **Duplicate the SIF**.

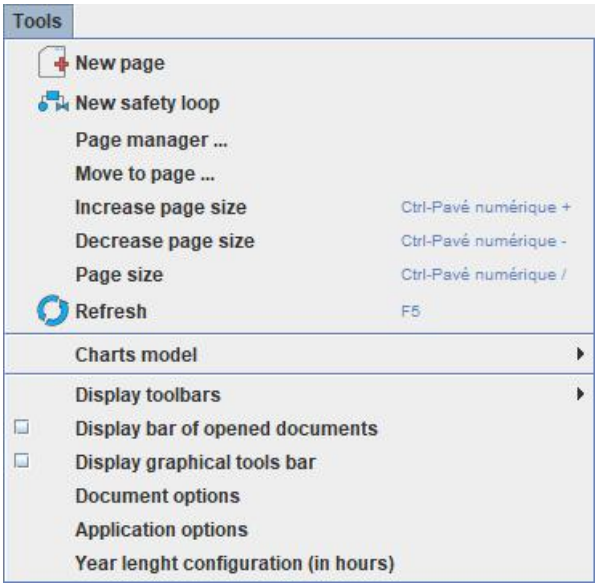
The functionality **Remove** deletes selected graphic elements.

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

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

6.3. 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 **New safety loop** action creates a new Safety Instrumented Function. A new page will be added if the current page contains a SIF already.

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 14, “Zoom and page size”

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

The function **Charts model** is used to modify the default curves displayed for a SIF.

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 **Display graphical tools bar** checkbox enables/hides the vertical tools bar on the left.

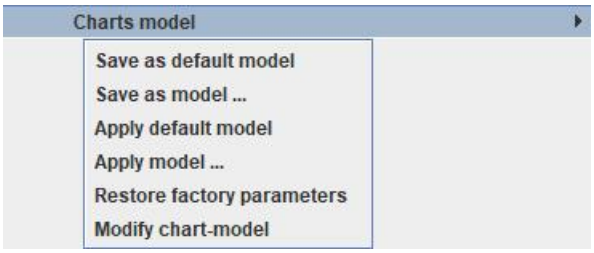
The functionality **Document options** : Opens a window to configure the document options. You have the possibility to configure a very large number of GRIF-Workshop's features (cf. Section 1, “Options of GRIF - SIL”). Some options only apply to the application and are accessible via the menu **Application 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 1, “Options of GRIF - SIL”

The functionality **Year length configuration (in hours)** allows you to change the number of hours in a year. The scope of this option is global to all GRIF modules.

6.3.1. Charts model

The function **Charts model** is used to modify the default curves displayed for a SIF.



The function **Save as default model** saves the current curves as default template for curves. New SIF will be added with this curve template.

The function **Save as model ...** saves the current curves in a model file that can be used by **Apply model ...**.

The function **Apply default model** restores curves to the default template.

The function **Apply model ...** applies the selected curve template file (created with **Save as model ...**).

The function **Restore factory parameters** resets any modification on curve and restore the default model of the software when it was installed.

The function **Modify chart-model** opens the configuration window of the selected curve.

6.4. Document

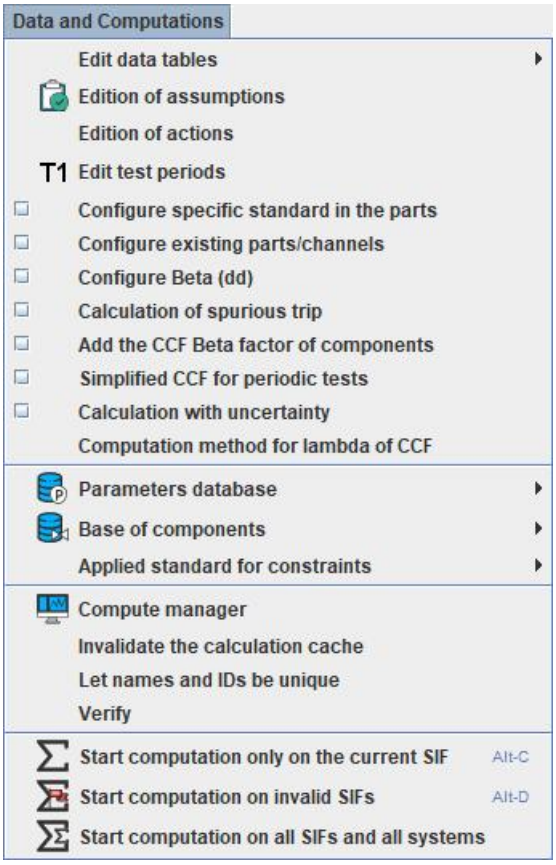
The menu **Document** gives access to all documents being modified or produced.

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

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

6.5. 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.

Edition of assumptions : Opens a new non-blocking window containing the editing table of assumptions. The assumptions' settings are detailed here Section 15, “Hypothesis”.

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

The **Edit test periods** action change the testing period of every component. This action is also available id the top toolbar.

The **Configure specific standard in the parts** action allows configuration of a specific standard for each part (actuators/sensors) of a SIF. Is not checked, default standard of the document will be used.

The **Configure existing parts/channels** action allows configuration of a channel (resp. a part) as being a channel (resp. a part) existing somewhere else. It displays this setting panel in architecture panel.

The **Configure Beta (dd)** allows CCF handling for dangerous detected failure using a Beta (dd).

The **Calculation of spurious trip** action enables spurious trip are calculation.

The **Add the CCF Beta factor of components** is for advanced users. If a component is used in several channels or parts defining each CCF Beta factor:

- if checked, component will be added in all the CCFs impacting it.
- if unchecked, component will be connected only to the "original" CCFs of the component (ie. not defining existing component).

If **Simplified CCF for periodic tests** action is unchecked, when several "Periodically tested" components are impacted by a CCF,the worst parameters are chosen for the CCF.If the option is checked then Gamma, Pi, X, Sigma, and Omegas are set to 0 in the CCF law,moreover repair will be instantaneously done.

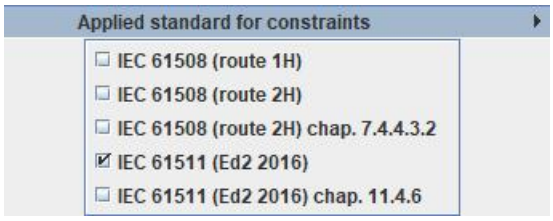
The **Compute the uncertainties** allows, if selected, to take into account the uncertainties during the computations.

The **Computation method for lambda of CCF** opens a windows to configure the way to choose the lambda of "Beta TOTAL" CCF. This feature is detailed in Appendix A, *Configuration of Lambda computation method for CCF*.

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 8.1, “Database of parameters”.

The **Base of components** action handles the connection to an *.xlsx file where components are stored.

The **Applied standard for constraints** action is used to define which standard will be used to compute the maximum SIL due to architectural constraints. The different standards are explained in Architectural constraints definition.



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 9.3, “Compute manager”.

The functionality **Invalidate the calculation cache** : To optimize calculations, some calculations data are cached. Invalidate calculation cache allows you to completely empty these data and ensure authentic results. In normal use of the software, it is not necessary to use this function.

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.

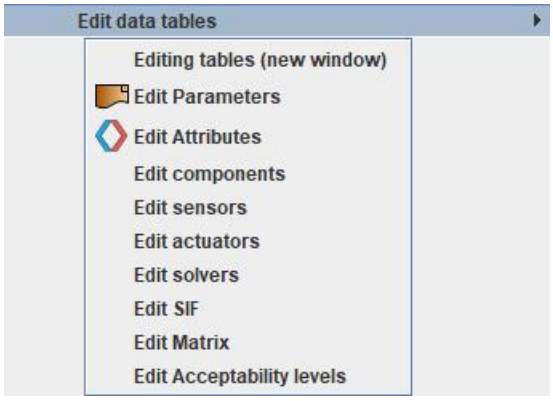
The **Start computation only on the current SIF** action starts computations, please see Section 9.1, “Launch PFD/PFH computation” for more details.

The **Start computation on invalid SIFs** action computes only SIF that are invalidated. A SIF is invalidated if it as been modified since last computation, either directly (modification of the SIF itself) or indirectly (modification of a component in another SIF but used be the SIF)
Only SIF that need recomputation are really recomputed.

The **Start computation on all SIFs and all systems** action starts computation on every SIF and every system of the document.

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

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

The **Edit components** action opens a window displaying a table containing all components of the document. Parameters that are common to solvers, sensors and actuators can be modified (see Section 4.5, “Edition of data table”).

The **Edit sensors** action opens a window displaying a table containing all sensors, their configuration is detailed in Sensor(s).

The **Edit actuators** action opens a window displaying a table containing all actuators, their configuration is detailed in Actuator(s).

The **Edit solvers** action opens a window displaying a table containing all solvers, their configuration is detailed in Solver.

The **Edit SIF** action opens a window displaying a table containing all SIF, some parameters can be modified and computation results are displayed.

Edit Matrixes : Opens a non-blocking window containing the matrixes editing table. Matrixes settings are detailed here Section 12.2, “Entering risk matrix models”.

Edit Acceptability levels : Opens a non-blocking window containing the acceptability levels editing table. Acceptability levels settings are detailed here Section 12.1, “Entering matrix acceptability levels”.

6.5.2. 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 8.1, “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.

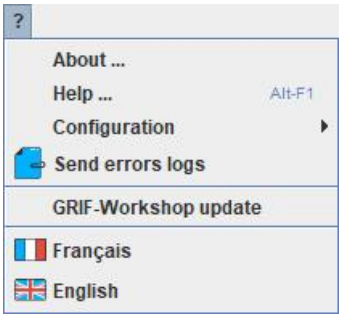
The functionality **Rebuild links to the database** : Attempts to reconnect parameter's settings of a document to data from the database Opens a window that highlights parameters that can be reconnected.

6.6. Add-ons

Depending on the version, some GRIF add-ons can be added to this menu.For more details on the add-ons, refer to the add-on's documentation.

6.7. ?

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.

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.

6.7.1. Configuration

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

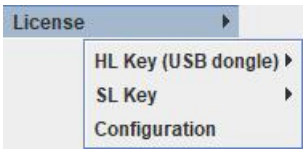
The menu **Licence** groups the configuration functions of the license server. For more information on the use of licenses, please refer to the GRIF installation manual.

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.

6.7.1.1. 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.

6.7.1.1.1. 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.

6.7.1.1.2. 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.

6.7.1.2. Associate GRIF files

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

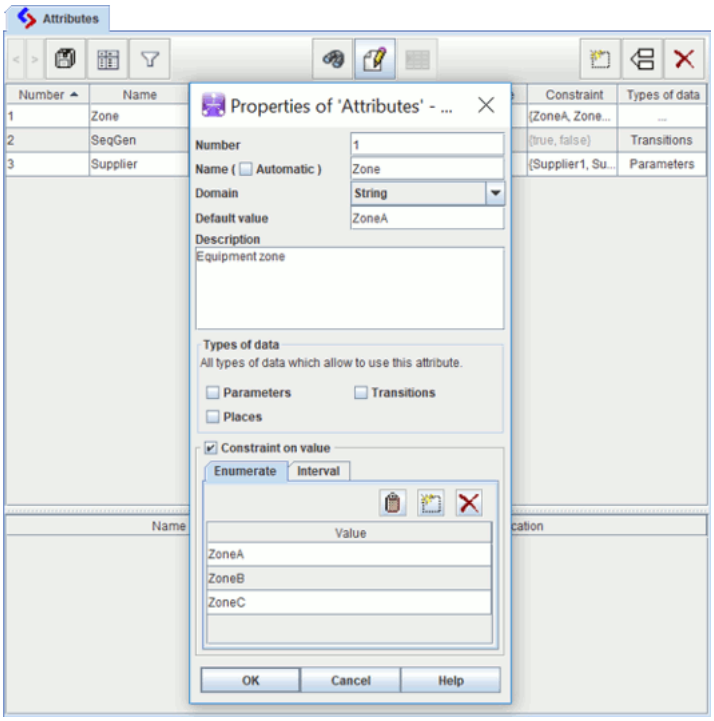
The functionality **For current user** : Associates GRIF files to the current user

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

7. Attributes

7.1. 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:

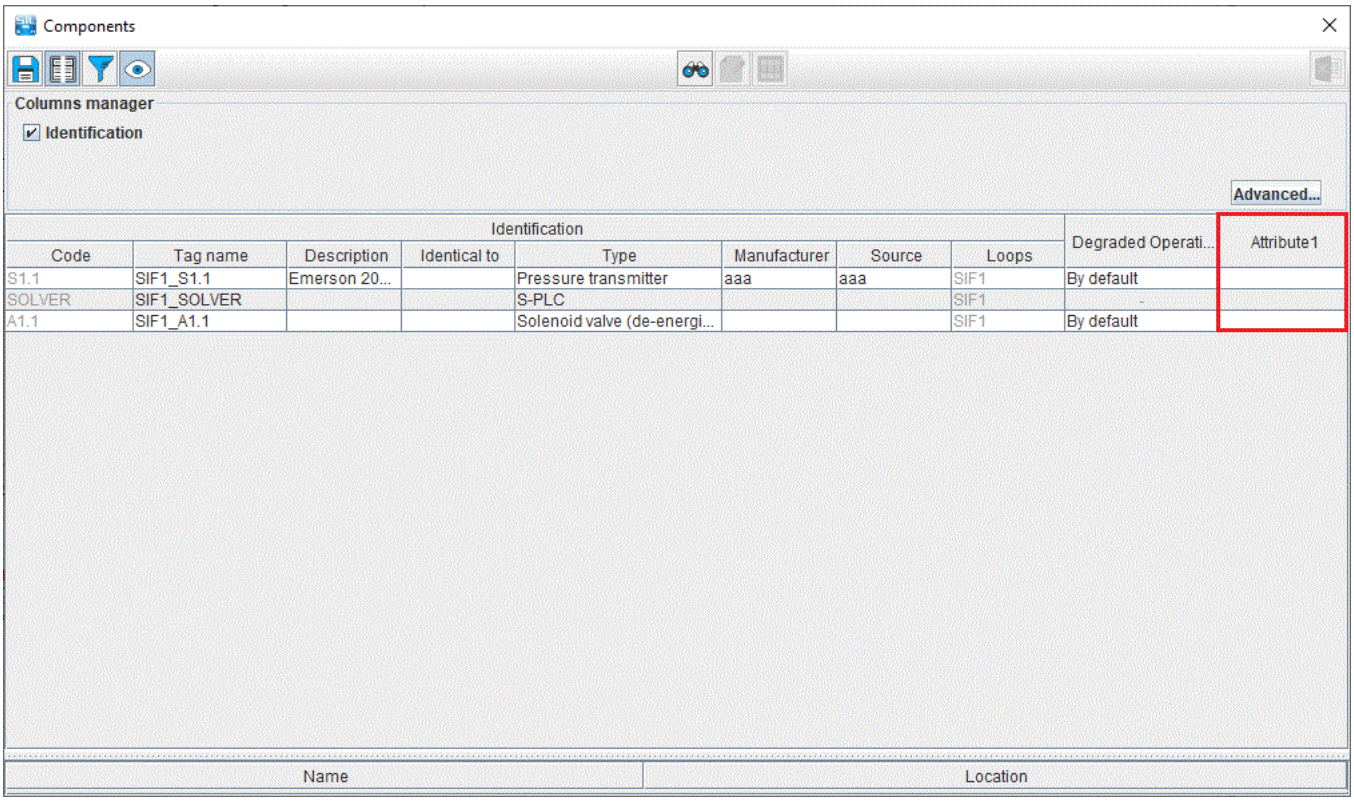
- **boolean**: This kind of attribute is a boolean;
- **integer**: This kind of attribute is used to affect an integer value;
- **float**: This kind of attribute is used to affect a float value;
- **string**: This kind of attribute is used to affect a free text.

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**.

7.2. Use of the attributes

In a SIF loop, it is possible to associate attributes on components directly on the components tables :



Attributes of a component can be imported from / exported to a component database. To import a component with an attribute from a database, the attribute must not have a reserved name, else an error will occur.

8. Data Bases

8.1. 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).

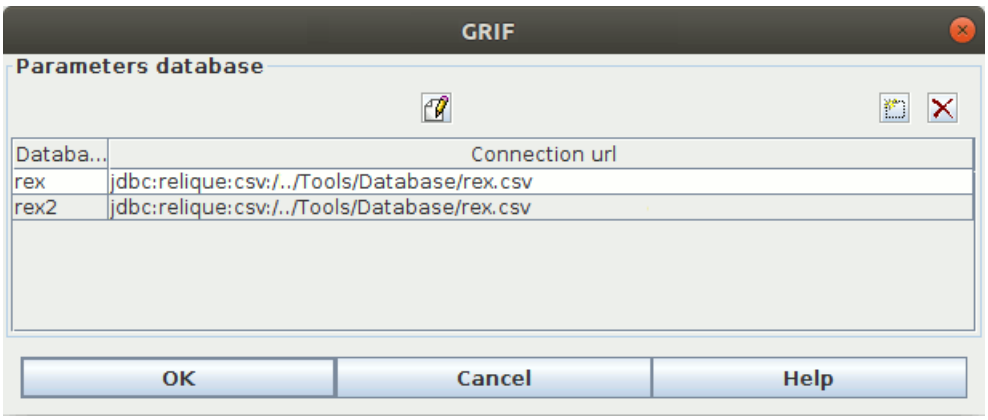
8.1.1. Format of the databases

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

8.1.2. 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.

8.1.2.1. Connection to a CSV file

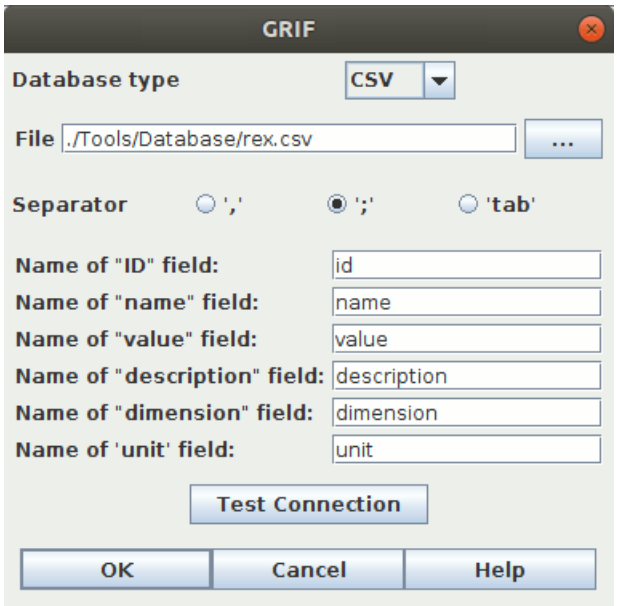
8.1.2.1.1. Form of the database

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
```

8.1.2.1.2. 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.

8.1.2.2. Connection to a XLS file

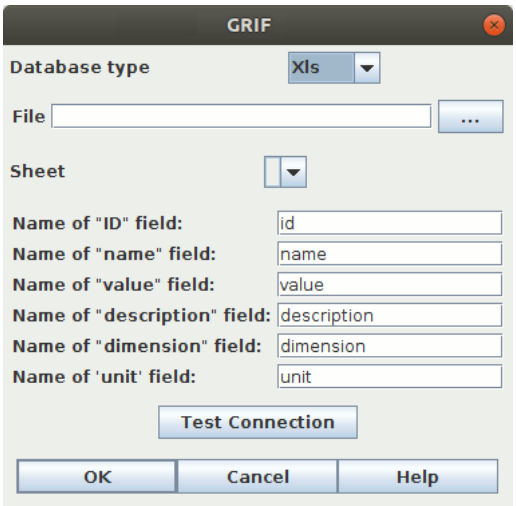
8.1.2.2.1. Form of the database

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						

8.1.2.2.2. Connection

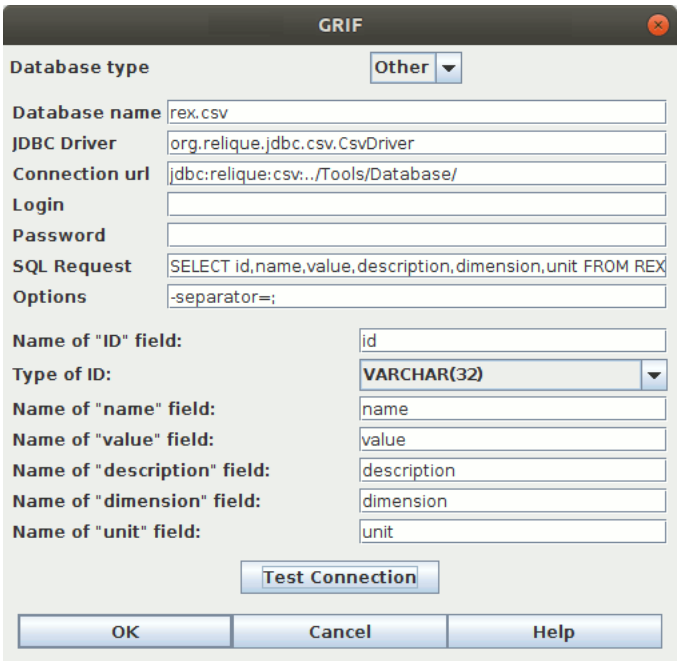
To connect GRIF to this database, select the **XLS** type in the connection window. The window is now as followed:



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.

8.1.2.3. 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.

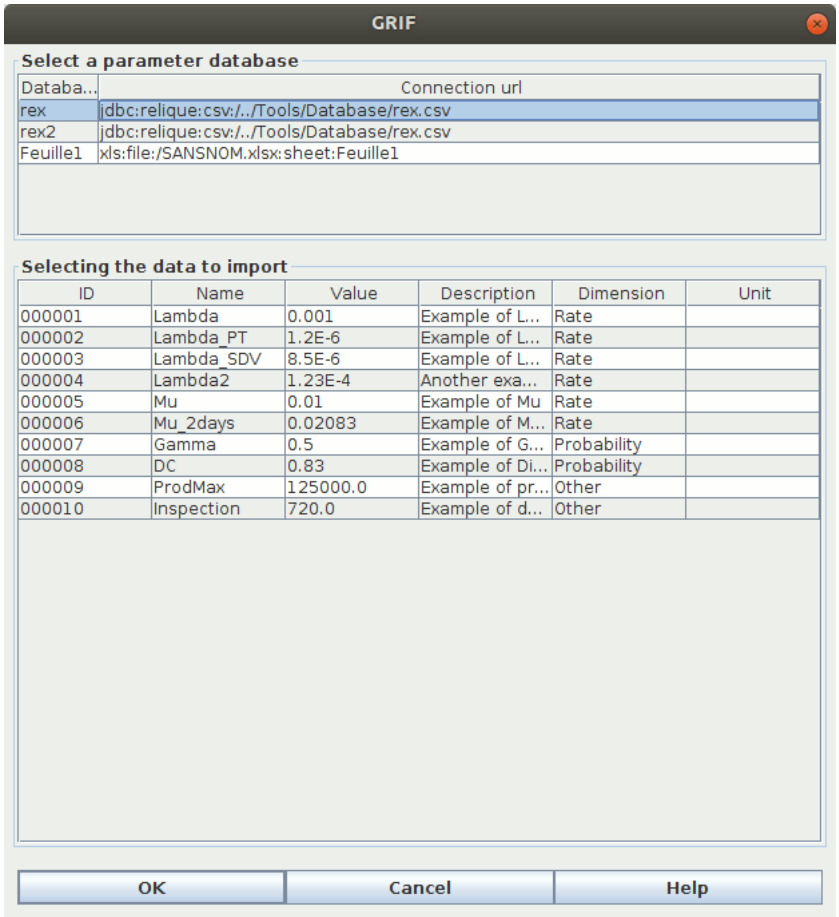
4. The SQL request **SELECT id,name,value,description,dimension,unit FROM REX** is used to gather the dates.

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.

8.1.3. 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.

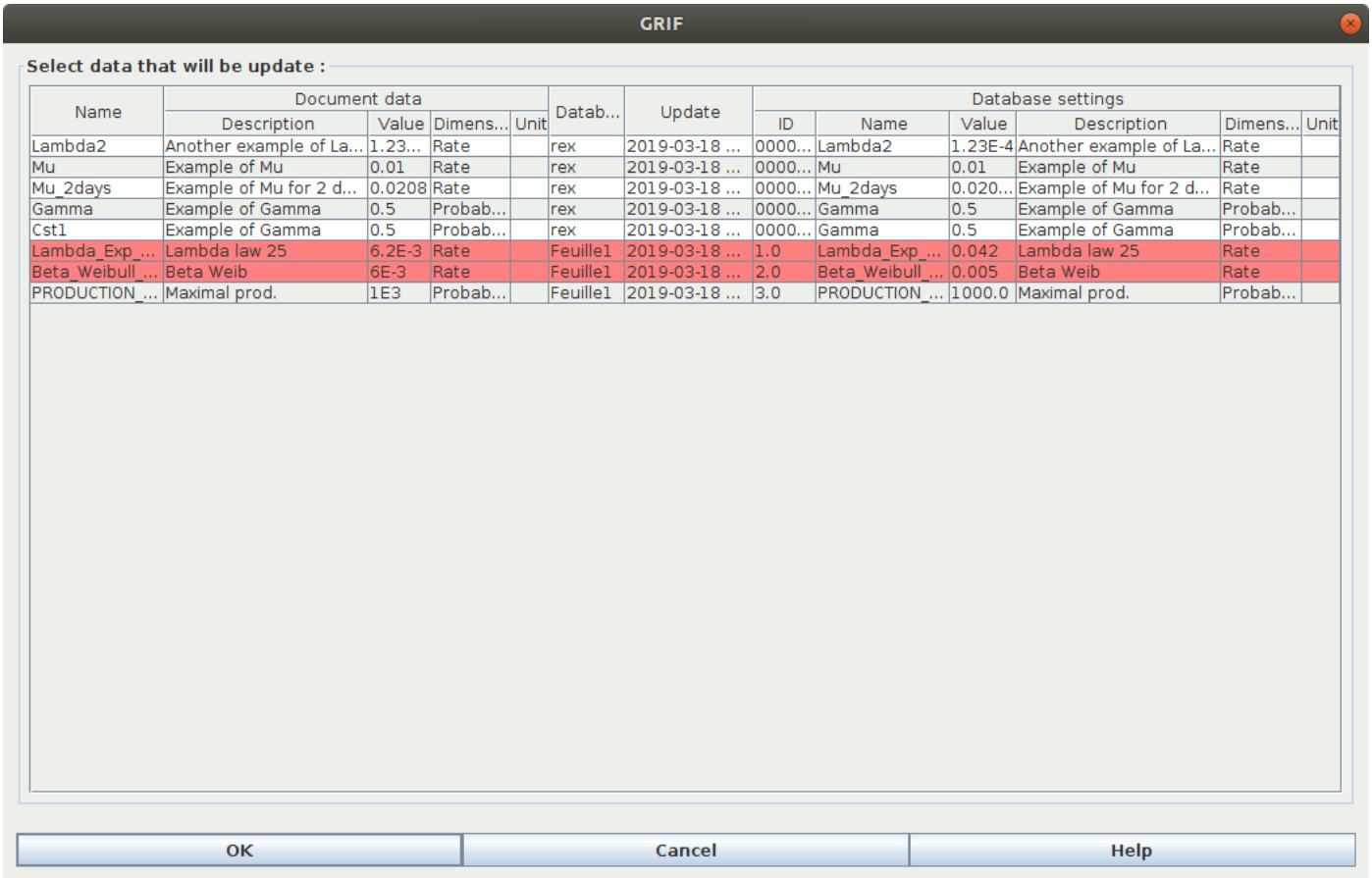


Select the parameters you want to import, and click on **OK**. The parameters are now created and imported in GRIF. The created parameters have the same names than the database's parameters, and the fields "Description" or "Dimension" are identical of those found in the database.

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.

8.1.4. 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:

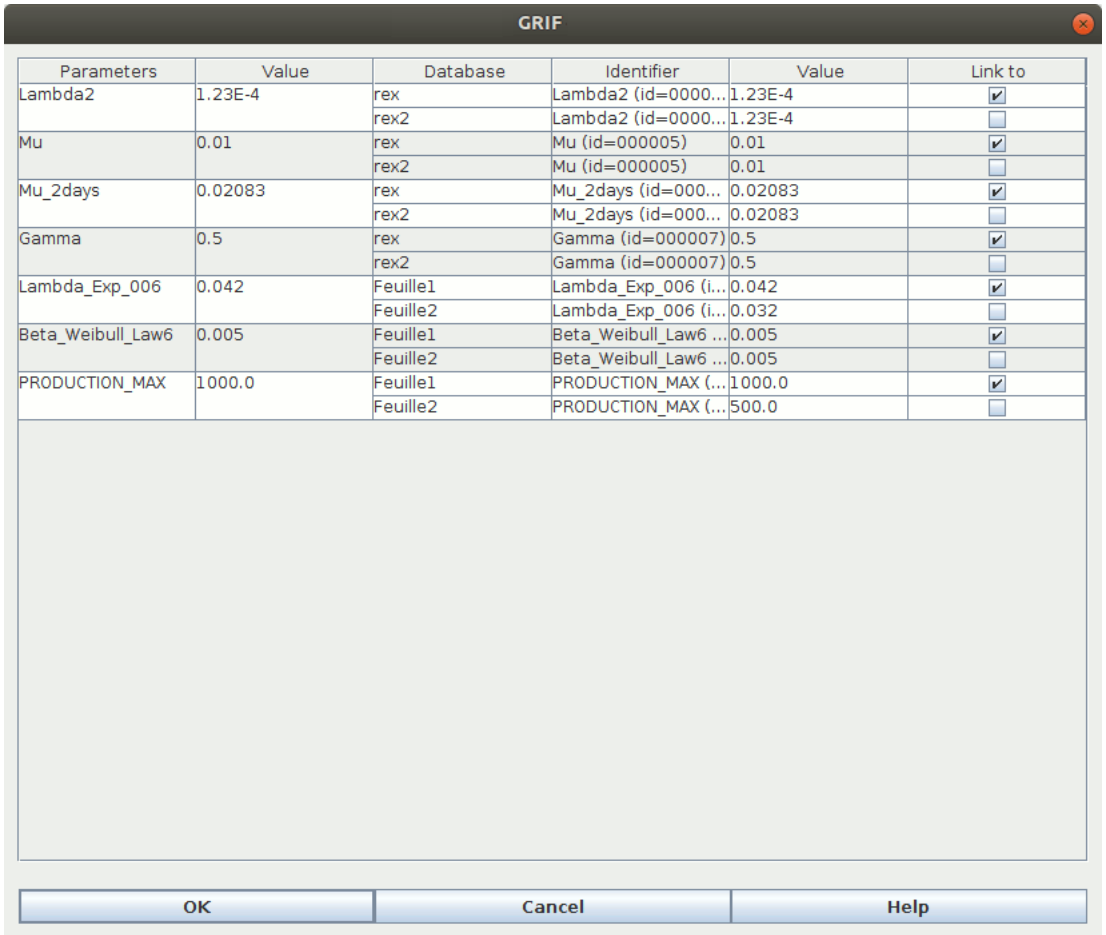


Name	Document data				Datab...	Update	Database settings					
	Description	Value	Dimens...	Unit			ID	Name	Value	Description	Dimens...	Unit
Lambda2	Another example of La...	1.23...	Rate		rex	2019-03-18 ...	0000...	Lambda2	1.23E-4	Another example of La...	Rate	
Mu	Example of Mu	0.01	Rate		rex	2019-03-18 ...	0000...	Mu	0.01	Example of Mu	Rate	
Mu_2days	Example of Mu for 2 d...	0.0208	Rate		rex	2019-03-18 ...	0000...	Mu_2days	0.020...	Example of Mu for 2 d...	Rate	
Gamma	Example of Gamma	0.5	Probab...		rex	2019-03-18 ...	0000...	Gamma	0.5	Example of Gamma	Probab...	
Cst1	Example of Gamma	0.5	Probab...		rex	2019-03-18 ...	0000...	Gamma	0.5	Example of Gamma	Probab...	
Lambda_Exp_...	Lambda law 25	6.2E-3	Rate		Feuille1	2019-03-18 ...	1.0	Lambda_Exp_...	0.042	Lambda law 25	Rate	
Beta_Weibull_...	Beta Weib	6E-3	Rate		Feuille1	2019-03-18 ...	2.0	Beta_Weibull_...	0.005	Beta Weib	Rate	
PRODUCTION_...	Maximal prod.	1E3	Probab...		Feuille1	2019-03-18 ...	3.0	PRODUCTION_...	1000.0	Maximal prod.	Probab...	

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.

8.1.5. 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.

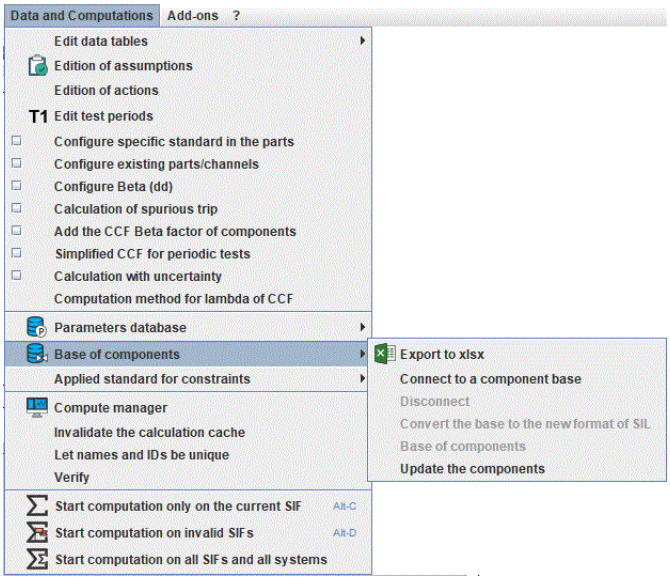


Parameters	Value	Database	Identifier	Value	Link to
Lambda2	1.23E-4	rex	Lambda2 (id=0000...	1.23E-4	☑
		rex2	Lambda2 (id=0000...	1.23E-4	☐
Mu	0.01	rex	Mu (id=000005)	0.01	☑
		rex2	Mu (id=000005)	0.01	☐
Mu_2days	0.02083	rex	Mu_2days (id=000...	0.02083	☑
		rex2	Mu_2days (id=000...	0.02083	☐
Gamma	0.5	rex	Gamma (id=000007)	0.5	☑
		rex2	Gamma (id=000007)	0.5	☐
Lambda_Exp_006	0.042	Feuille1	Lambda_Exp_006 (i...	0.042	☑
		Feuille2	Lambda_Exp_006 (i...	0.032	☐
Beta_Weibull_Law6	0.005	Feuille1	Beta_Weibull_Law6 ...	0.005	☑
		Feuille2	Beta_Weibull_Law6 ...	0.005	☐
PRODUCTION_MAX	1000.0	Feuille1	PRODUCTION_MAX (...	1000.0	☑
		Feuille2	PRODUCTION_MAX (...	500.0	☐

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.

8.2. Bases of components

Users can create base of components with all characteristics needed to create safety loops in an Excel file. This feature is available in the menu **Data and Computations - Bases of components** .



8.2.1. Export to xlsx

This action enables to create an Excel file with the correct format to be used for the creation of the component base which will then be compatible with the SIL module.

This database contains by default the equipment present in the current document when it is created and can be modified to create a database with your own data.

The file created has 3 tabs for each part of the the architecture :

- **SENSORS** ;
- **ACTUATORS** ;
- **SOLVERS** ;

D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
RUMENTED_TYPE	MANUFACTURER	DATA_SOURCE	DETERMINED_CHARACTER	TEST_TYPE	T0	T0_UNIT	T1	T1_UNIT	MODE	LAMBDA	DFF	DC_D	DC_S	LAMBDA
ISMITTER_FLOW			F	TESTUNITSTOP	6.0	YEAR	6.0	YEAR	FACTORISED	1.5E-6	25.0	70.0	100.0	1.125E-7

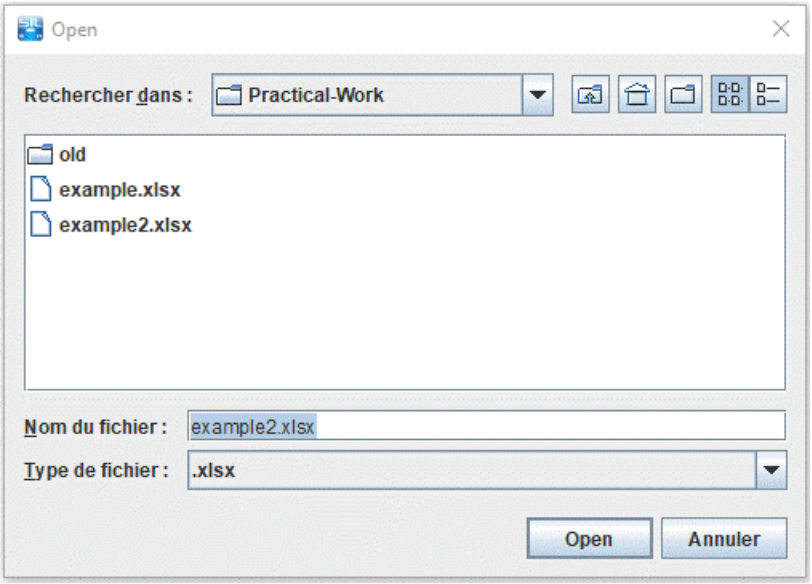
UATORS | SOLVERS

All the information required for the creation of a component can be filled in the database. The headings of the columns are indicated in the chapter "Configuration of components"

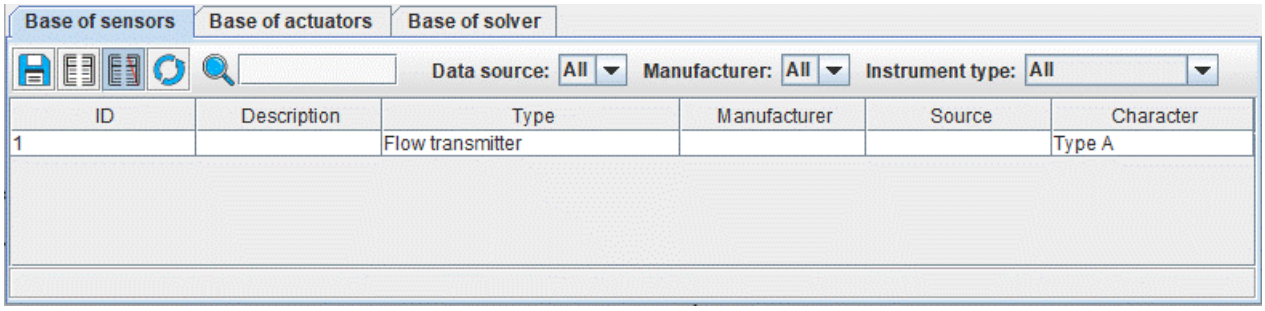
⚠ IIt is not necessary to create all the columns in the Excel file. In this case, the values of the information of the elements not present will be the default values of the components.

8.2.2. Connect to a component base

This action enables, once the database file has been created, to connect it to the SIL Module. A window opens and enables you to search the file :



After selecting the correct file the following window opens and enables to display the database information



8.2.3. Disconnect

This action enables to disconnect from the database currently linked to the SIL module

8.2.4. Base of component

This action enables to visualize the connected database with the same database information visualization window dispalyed with **Connexion à une base de de composant** action

8.2.5. Update the components

This action enables to update components linked to a database if it has been modified

8.2.6. Convert the base to the new format of SIL

This action enables to update of the database if in the versions of GRIF subsequent to its creation modifications have been made in its structure.

9. Computes

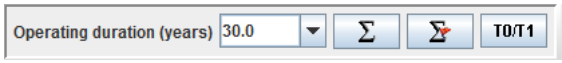
9.1. Launch PFD/PFH computation

When all of the components have been configured, computations can be started. PFD and PFH Computations are performed for each SIF. User can choise between:

- launch calculation only in the current SIF: in this case, calculations are launched via the menu **Data and Computations / Start computation only in the current SIF** , or in the icon Σ
- lauch computation on invalided SIF: in this case, calculations are launched via the menu **Data and Computations / Start computation on invalided SIF** , or in the icon Σ . All modified SIF by the user will be calculate.

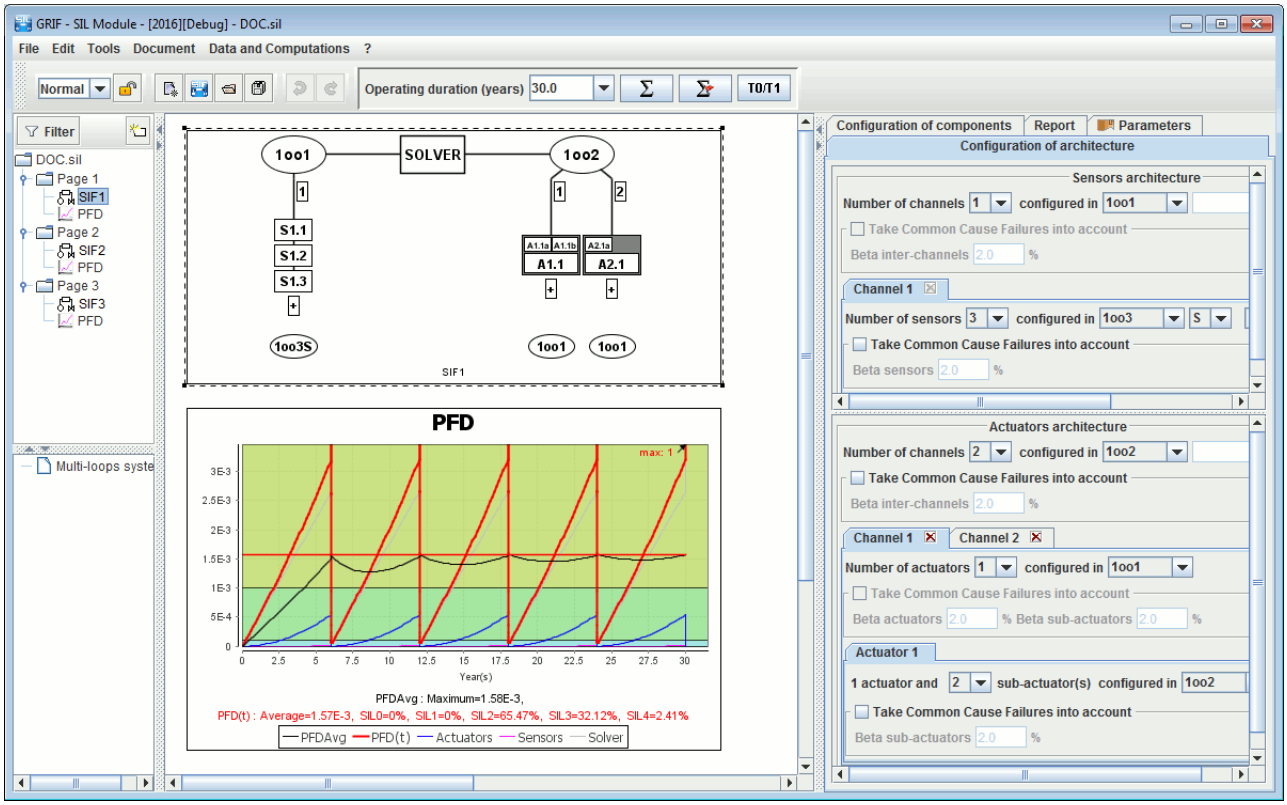
  Invalided SIF are highlighted with the following icon:

All these launch computation commands are grouped in the icon bar:



9.2. Computations results

When computation is performed, the chart, which contained no information, has now been updated.



The combination Control+Scroll wheel enables you to enlarge (zoom in) or reduce the window.

The x-axis represents the time in hours and the y-axis represents the probability of failure of the SIF when triggered, also called PFD. The chart ranges from 0 to 30 years by default but it is possible to modify this value as explained in the chapter on curves. There are 5 curves in the chart:

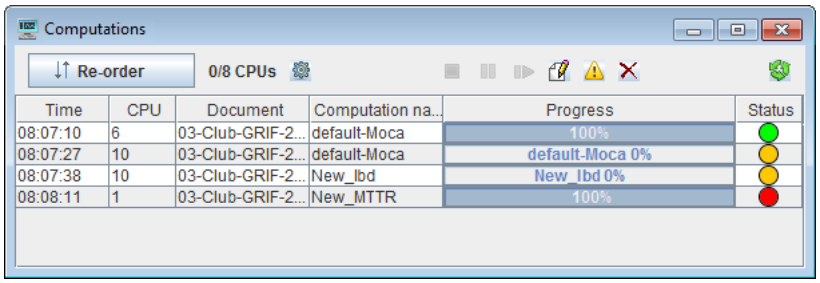
- **PFD(t) or PFH(t)**: the instantaneous value of the system's PFD/PFH.
- **PFD Avg or PFH**: the average value of the system's PFD/PFH.
- **Actuators**: the instantaneous value of the PFD/PFH of the actuator part of the system.
- **Sensors**: the instantaneous value of the PFD/PFH of the sensor part of the system.
- **Solver**: the instantaneous value of the solver's PFD/PFH.

The curves are located in one or several bands of color. These bands represent the PFD ranges, which define the SIL:

- **SIL 0**: instantaneous PFD ∈ [10^{-1}; 1]. instantaneous PFH ∈ [10^{-5}; +infinity].
- **SIL 1**: instantaneous PFD ∈ [10^{-2}; 10^{-1}[. PFH instantanée ∈ [10^{-6}; 10^{-5}[.
- **SIL 2**: instantaneous PFD ∈ [10^{-3}; 10^{-2}[. instantaneous PFH ∈ [10^{-7}; 10^{-6}[.
- **SIL 3**: instantaneous PFD ∈ [10^{-4}; 10^{-3}[. instantaneous PFH ∈ [10^{-8}; 10^{-7}[.
- **SIL 4**: instantaneous PFD ∈ [0; 10^{-4}[. instantaneous PFH ∈ [0; 10^{-8}[.

9.3. Compute manager

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

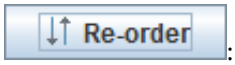


Compute manager is automatically displayed when calculations are performed. User can display the window using the following icon .

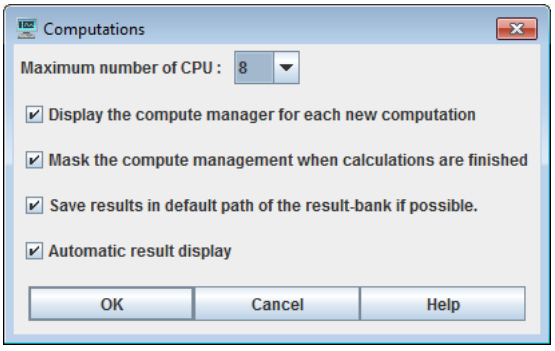
This tab is made of 6 columns:

- **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.

10. Multi-loop systems

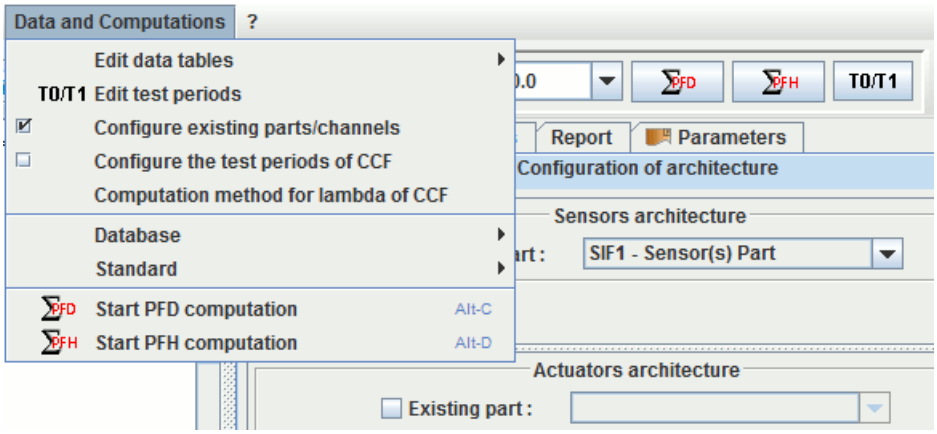
10.1. Creation of sevaral instrumented loops

It is possible to create several instrumented loops in a same document. To add new SIF, use the following icon .

New loops are configured in the same way as the first one (cf. **Configuration of components**).

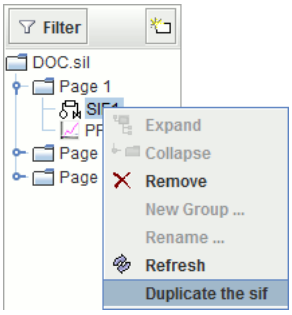
Different SIF created can use parts or components already present in others loops:

- **an existing component** .The existing component can be selected in a list. It can be a component of the current SIF, or one of another safety instrumented loop. This options is only available when you have many components of the same type.
- **an existing part** . In this case, it is all sensor-parts or actuartor-parts which are identicals. To use this functionality, check **Configure existing Parts/Channels** in **Data and Computations** . After, in Actuartor-part or Sensor-part, check **Existing part** and select reference part.

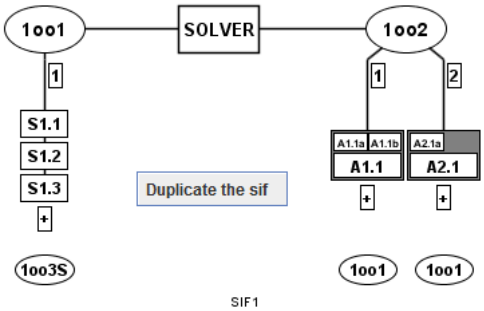


A new safety instrumented loop can be created by a duplication of another one already created.

With a right click on SIF to duplicate in the arborescence:

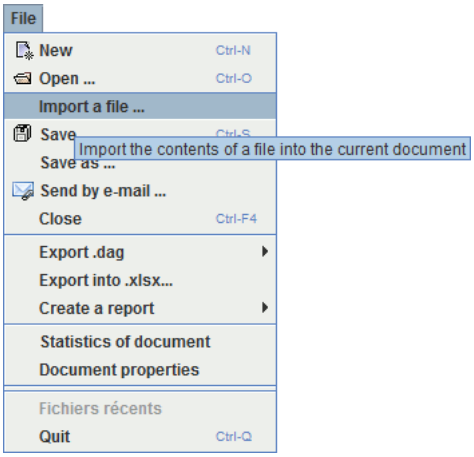


With a right click on the SIF directly in the graphical imput zone:

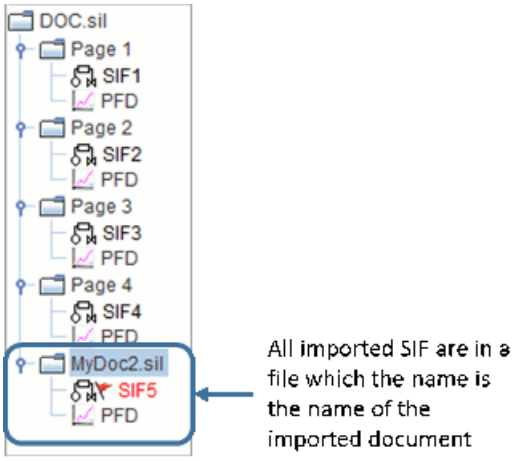


10.2. Import from a another document

It is possible to add new SIF with an import from another document. In **Files** menu, select **Import a file** .

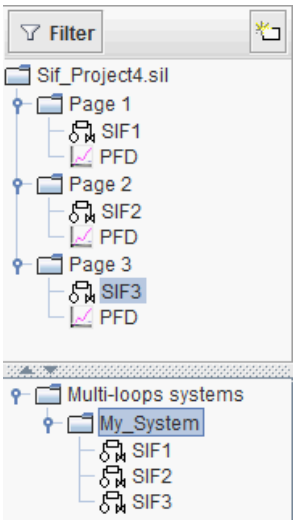


In this case, imported SIF will be in a file which the name is imported file:



10.3. Presentation

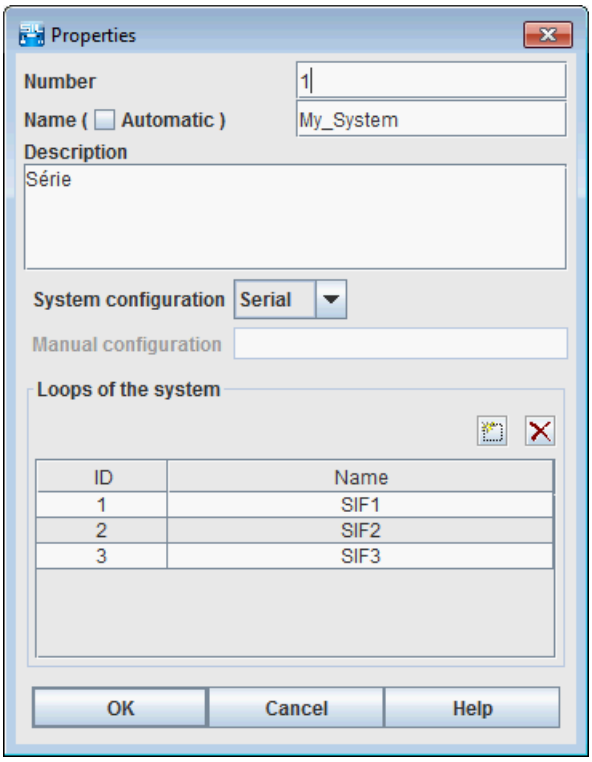
When several loops (SIF) are created in a document, a **tree view** on left left become visible. The upper part let you browse the pages (one page by SIF). The lower part is for systems that are made with several loops.



10.4. Input

Double-click one the root of **Multi-loops systems** tree in order to create an empty system.

System editing can be done either with a double-click or with a right-click using **Properties** menu. The following window is displayed:



You can enter **Number** and a **Name** . The **Automatic** checkbox generates a name starting with a base name followed by the number.

A text area is available for adding a **Description** to your system.

The **System configuration** defines the logical use of the loops.

- **Serial** : Safety loops are in serial, every loop must be available to have an available system.
- **Parallel** : Safety loops are in Parallel boucles de sécurité sont en parallèle, le système reste disponible tant qu'il reste au moins une boucle disponible.
- **Manual**: You can specify the configuration of the system as 1 & (2 | 3), with 1,2 et 3 corresponding to the loops which ID is 1 2 and 3.

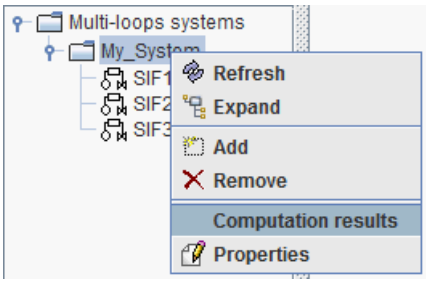
The **Loops of the system** part enables adding/removing/modifying of loops that are in the system.

10.5. Computations

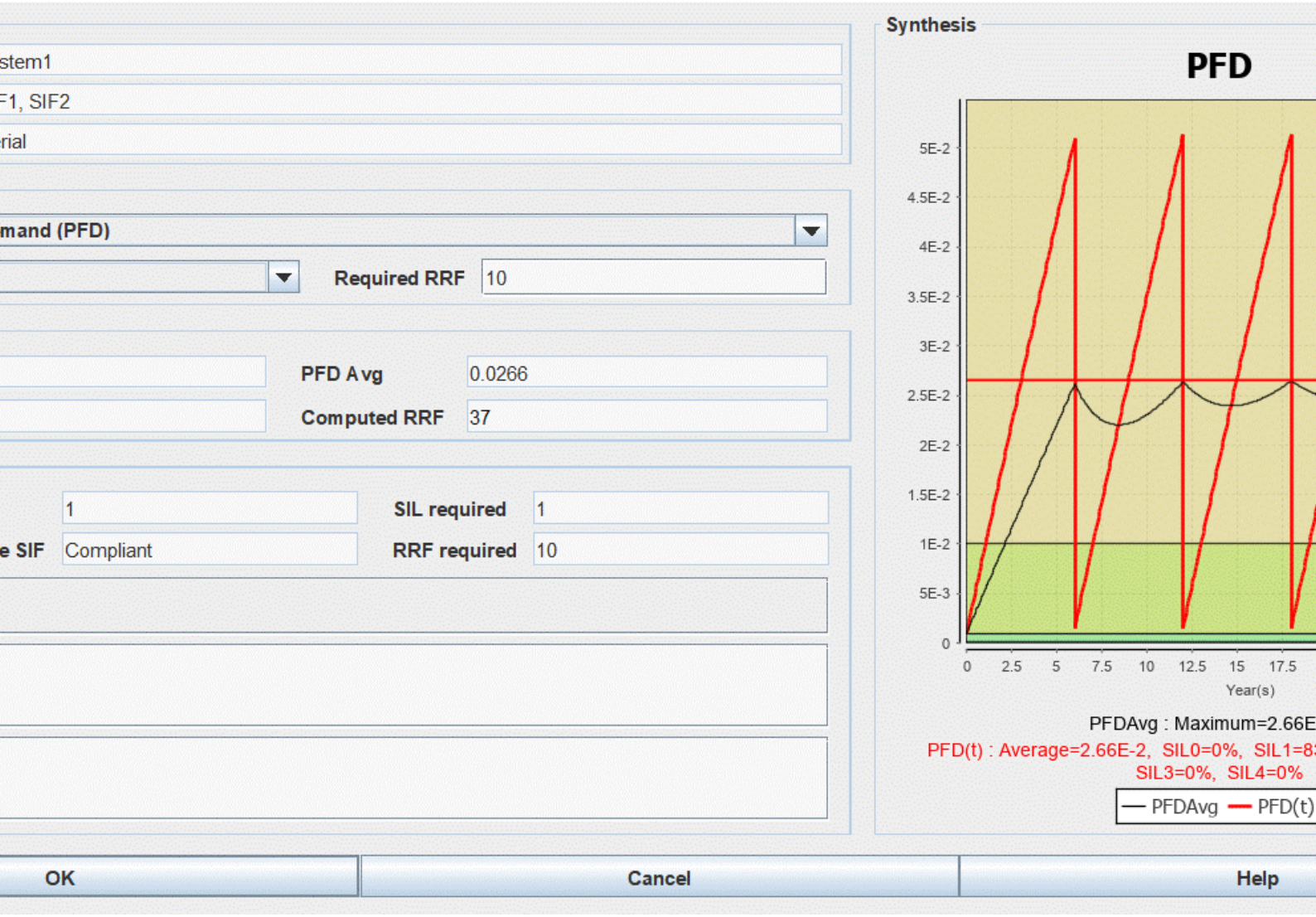
System PFD/PFH computations are made as for SIF PFD/PFH computations. (Section 9, “Computes”)

10.6. Reports and results

Results are available with a right-clic on the system, and then select **Computation results** in the menu.



The following window is displayed:



The **Description** part sum system configuration up.

The **For the system** part is used to specify the target values:

- **Requiered SIL** : value of the SIL that is requiered for the system.
- **Requiered RRF** : value of the RRF that is requiered for the system.

The **Computation** part displays computed values:

- **Operating duration** : the operating duration used for the computation.
- **PFD or PFH** : The computed PFDAvg or PFH.
- **Computed SIL** : SIL computed from PFD or PFH (architectural constraints are not taken into account).
- **Computed RRF** : RRF computed from PFD or PFH .

The **Results** part displayed achieved targets:

- **Achieved SIL** : Identical to computed SIL.
- **Conclusion of SIL for the system** : conclusion (compliant or not compliant).
- **Remark** : Remark generated by software.
- **Comments** : User descriptions.

The **Synthesis** part displays PFD/PFH curves of the system

10.7. PDF report and MS Excel report

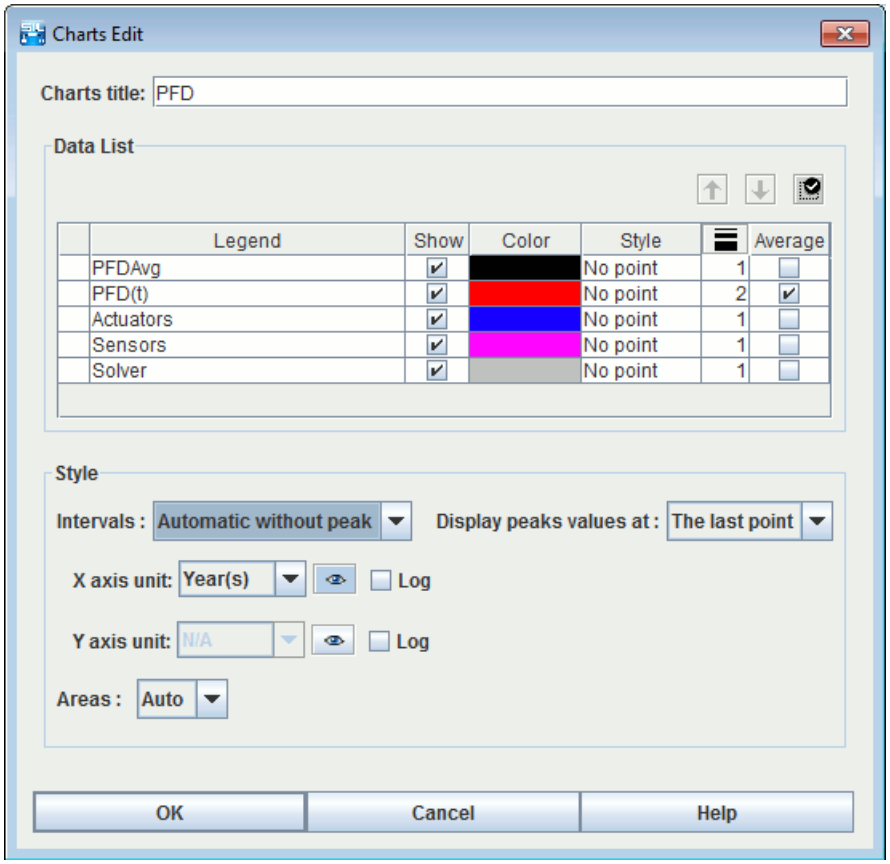
If there are system in your document, they will be automatically exported in reports. An additional section will be added for each system, at the end of the report. The system section contains information in **Reports and results** window.

11. Charts

Curves are drawn to study the results better. Five curves are available: PFDAvg(t), $PFD_{System}(t)$, $PFD_{Actuators}(t)$, $PFD_{Solver}(t)$, $PFD_{Sensors}(t)$.

11.1. Charts Edit window

The Charts Edit window is displayed when user double-click on charts.



This window is divided into several parts:

1. **Charts Title**: enables you to give a title to the graphic.

2. **Data List**: This part contains a three-column table listing the chart's different curves (name, description, display, curve colour, curve style, curve thickness).Several buttons are available above this table.
   - **Up** : moves the selected curve upwards in the list.
   - **Down** : moves the selected curve downwards in the list.
   - **Save as default model** : saves current chart setting as default setting for new documents.

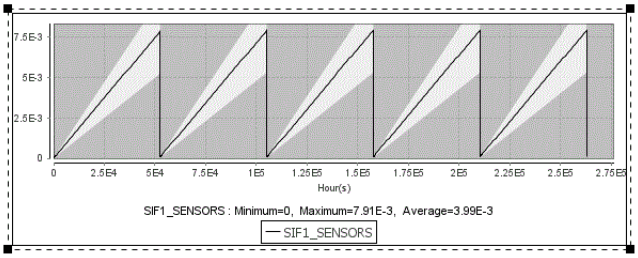
   For each curve, you can specify its colour, its style of points, its thickness and its display options.

3. **Style**: This part deals with displaying curves.
   - **Style type**: specifies the type of all the chart's curves (line or histogram). N.B: For histogram style, bars going outside drawing zone will be drawn with a gradient to warn user that he has to change intervals to see the entire bar.
   - **Intervals on X and Y**: Specifies the display interval for the X and Y axes (default interval or user-defined interval). This last function can, for example, be used to zoom in on the most interesting parts of the curve.

The **log** check boxes are used to enable the logarithmic scale on the axis concerned. Important: 0 cannot be represented on a log scale, remember to give a strictly positive starting point (e.g.: E-10). If 0 is given, the log scale will start with an arbitrary value E-15.

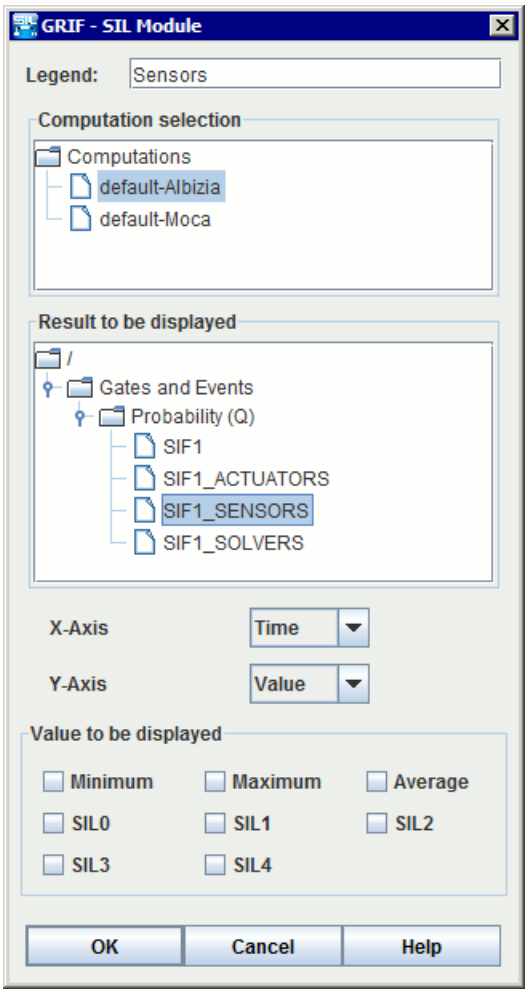
When domain axe deals with time, you can choose time unit among: hours, days, months, years. Default display is "hours" because it is the usualy used unit for modeling. It's only available in SIL module.

If the computations are with uncertainties, it is possible to check the **Display dispersion interval** and the chart will display the uncertainties interval.



11.2. Editing the curves

When a curve is edited (with a double-click on its name in list of curves), the curve edition window is displayed. The following window id displayed:



The window is divided into three parts:

- **Legends**: enables you to give a title to the curve.

- **Value to be displayed**: used to select the values which are to be displayed or not below the curve.

  For SIL curve (probability functions of times), available values are:
  - **SIL 0**: percentage of time spent in SIL 0.
  - **SIL 1**: percentage of time spent in SIL 1.
  - **SIL 2**: percentage of time spent in SIL 2.
  - **SIL 3**: percentage of time spent in SIL 3.
  - **SIL 4**: percentage of time spent in SIL 4.
  - **Minimum**: the minimum instantaneous PFD over the period studied.
  - **Maximum**: the maximum instantaneous PFD over the period studied.
  - **Mean**: the average of the PFD over the period studied.

  For average probability curves (like PFDAvg), available values are:
  - **Minimum**: the minimum value of the average PFD over the period studied.
  - **Maximum**: the maximum value of the average PFD over the period studied.
  - **Mean**: the average value of the average PFD over the period studied (which is NOT the PFDAvg).

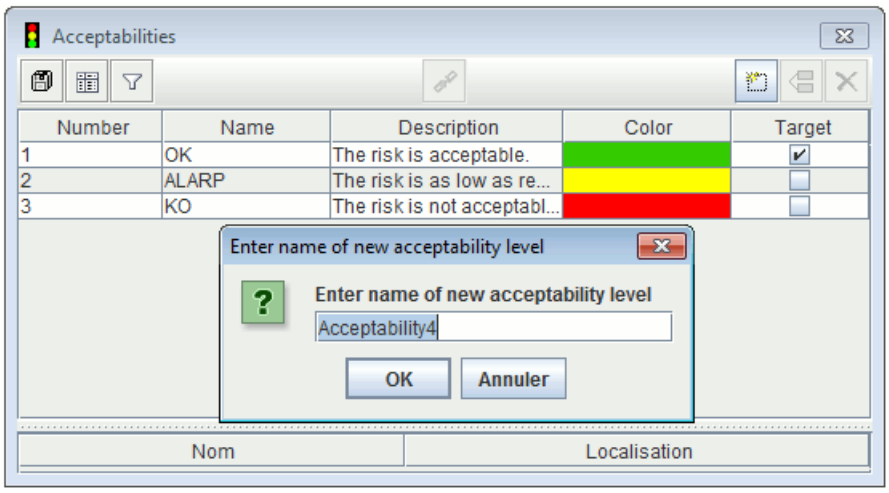
When the values are entered, just click on **OK** to close the windows.

12. Risk matrices

By default, new elements are initialized using a pre-constructed risk matrix. The default matrix can be modified or you can construct new ones. The following sections cover how to define and use risk matrices

12.1. Entering matrix acceptability levels

The levels of acceptability can be accessed only via the data edit tables. To enter a new **Acceptability level** , select the **Acceptability** tab in the data table and click on **Add** .



For further information on using data tables, please see ??? section.

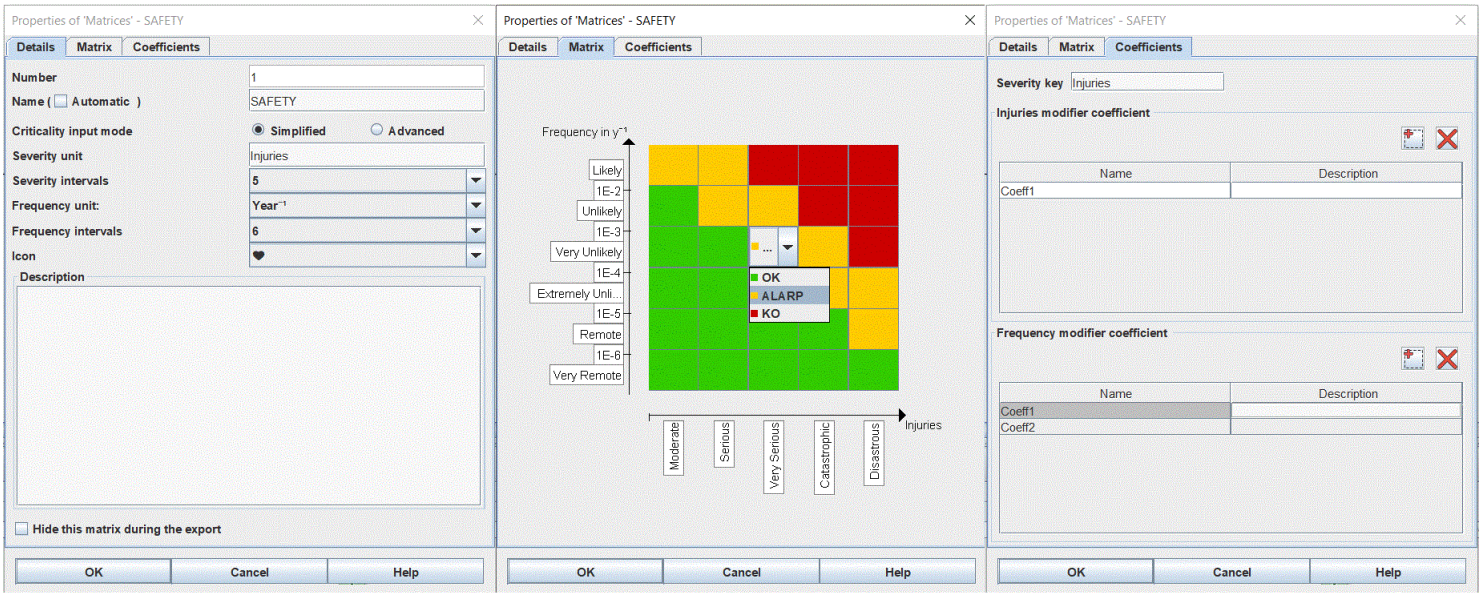
Each acceptability level is characterized by the following parameters:

1. A **number** : the number is an identifier. They are automatically incremented as and when new elements are created. This item of data is hidden from the user by default.
2. A **name** : this is a parameter that is defined automatically and can be modified by users. The default name comprises the "type"" followed by the "number" (e.g."Acceptability1").
3. A **description** : this field is used to add a description of the element. The purpose of this function is to make the model easier to read (by indicating the specificity of each element).
4. A **color** : this is the color that will be displayed in the risk matrix for this acceptability level.
5. A **target** : flag indicating whether or not the acceptability level is a target to be reached.

12.2. Entering risk matrix models

Risk matrices are used to graphically visualize the results of calculations run on consequences. TTo do so, at least one risk matrix model must be defined. Risk matrices can be accessed only via the data edit tables. To enter a new **Risk matrix model** , select **Risk matrix model** tab in the data table and click on **Add** .

For further information on using data tables, please refer to the section **???**



The **Details** tab is used to enter the following parameters:

1. A **number** : he number and type are the true identifiers for each element (which will be used by the computation engine). If a user wants to change the number of certain events, s/he must therefore make sure that no two events have the same number. They are automatically incremented as and when new elements are created..
2. A **name** : this is a parameter that is defined automatically and can be modified by users. The default name comprises the "type" followed by its "number" (eg: "MatrixModel1").
3. A **comment** : this field is used to add a description of the element. The purpose of this function is to make the model easier to read (by indicating the specificity of each element).
4. A **Criticity input mode**:
   - In **simplified** mode, the severity interval limits cannot be entered. The risk modifier coefficients are also hidden. The coefficients entered for the different consequences are identified from among all the risk intervals.
   - In **Advanced** the severity interval limits can be entered. The coefficients entered for the different consequences will be numerical values for all the severity intervals.
5. The **severity unity** :this value is displayed on the horizontal axis of the risk matrix.
6. The number of **severity intervals** : the number of columns displayed in the risk matrix.
7. The **frequency unit** : this value is displayed on the vertical axis of the risk matrix. Calculations are made in h^{-1} and converted into the chosen unit.
8. The number of **frequency intervals** : the number of lines displayed in the risk matrix.
9. The option **Hide this matrix during the export** : if this option is selected, the matrix will not be printing on the PDF export.

The **Matrix** tab is used to enter the following parameters:

1. **Acceptability levels** : if the acceptability levels have been correctly defined beforehand (see **Section 12.1, "Entering matrix acceptability levels"**) you can allocate an acceptability for each cell in the matrix by clicking on it and selecting the required acceptability level.
2. The axes: the names of the intervals and their limits can be modified by double-clicking on the text boxes. An increasing order must be respected on the axes.

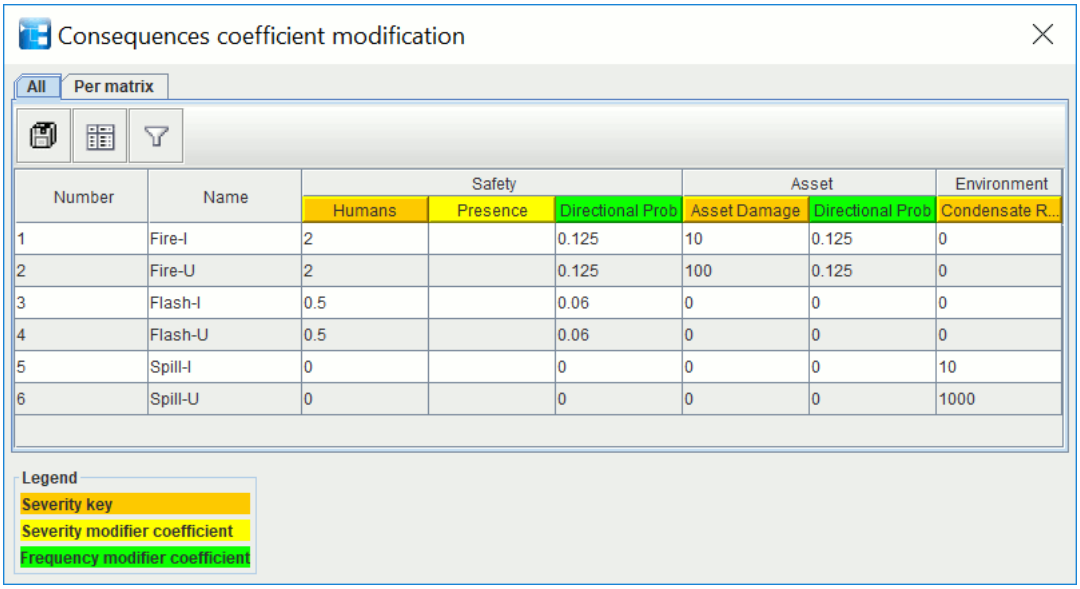
The **Coefficients** is used to define a set of frequency and risk modifier coefficients. The actual coefficient values must then be entered on the consequences (see **Section 12.3, "Entering coefficients"**)

1. **Severity key** this is the value affected by the risk modifier coefficients.
2. **Severity modifier coefficient** : these are used to reduce or increase the impact of the **Severity key** .
3. **Frequency modifier coefficient** : these are used to reduce or increase the frequency.

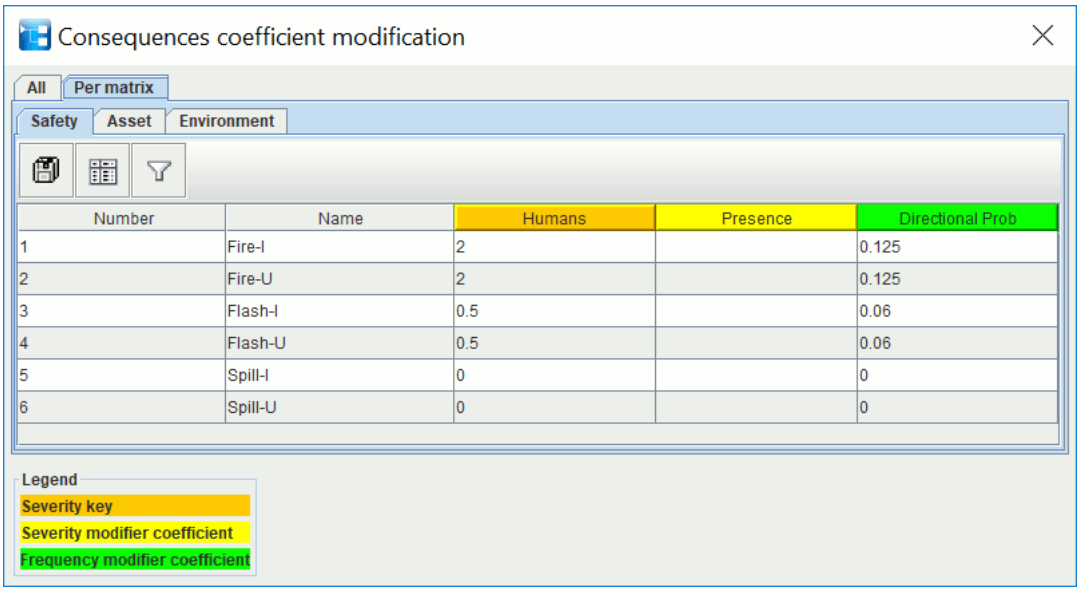
12.3. Entering coefficients

Coefficients are factors for (and which may or may not reduce) risks run or for the frequency of occurrence of a feared scenario. The frequency of occurrence of a feared scenario is determined based on the frequency of occurrences of the upstream scenarios. The risk of a feared event is obtained by multiplying the **Severity key** for the scenario by its **Severity modifier coefficient** .

The risk coefficients and keys are modeled on each risk matrix (see **Section 12.2, "Entering risk matrix models" [69]**) and allocated values for each feared event either via the scenario edit window (see **???**), or in the window below which can be accessed via the horizontal toolbar or via the menu **Data and Computations - Edit coefficients**



Number	Name	Safety			Asset		Environment
		Humans	Presence	Directional Prob	Asset Damage	Directional Prob	Condensate R...
1	Fire-I	2		0.125	10	0.125	0
2	Fire-U	2		0.125	100	0.125	0
3	Flash-I	0.5		0.06	0	0	0
4	Flash-U	0.5		0.06	0	0	0
5	Spill-I	0		0	0	0	10
6	Spill-U	0		0	0	0	1000



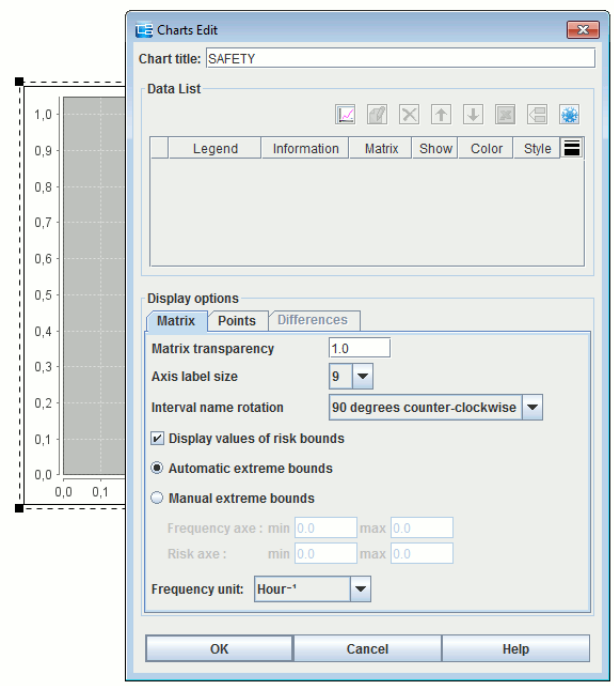
Number	Name	Humans	Presence	Directional Prob
1	Fire-I	2		0.125
2	Fire-U	2		0.125
3	Flash-I	0.5		0.06
4	Flash-U	0.5		0.06
5	Spill-I	0		0
6	Spill-U	0		0

The **All** tab gives edit access to all the keys and coefficients defined on the risk matrices for the entire document. The **Per matrix** tab gives edit access to the same data as for the **All** tab, but they are organized by matrix.

12.4. Risk matrix tool

This tools accessible since the vertical tool bar, enables to represent graphically the consequences through a model of risk matrix beforehand constructed. For more information on the construction of such a model, please refer to the section **Section 12.2, "Entering risk matrix models"** .

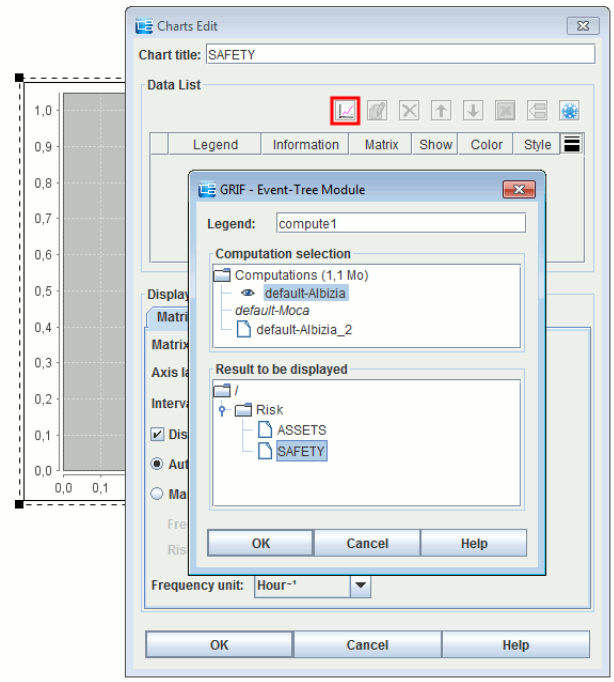
To use it, select tools then on the input zone, select the rectangle of display of the matrix. This space constitutes the zone of display of the risk matrix, it can be moved, deleted or modified the size. To configure the risk matrix, double-click on this zone of display. The following edition opens:



User can inform the following display parameters:

- **Chart title**: It is the title which will be shown above the risk matrix as well as in the tree graphic.
- **Data list**: It is computation results to use to show the consequences in the matrix.
- **Display options** : Configuration panels of the display of the matrix.

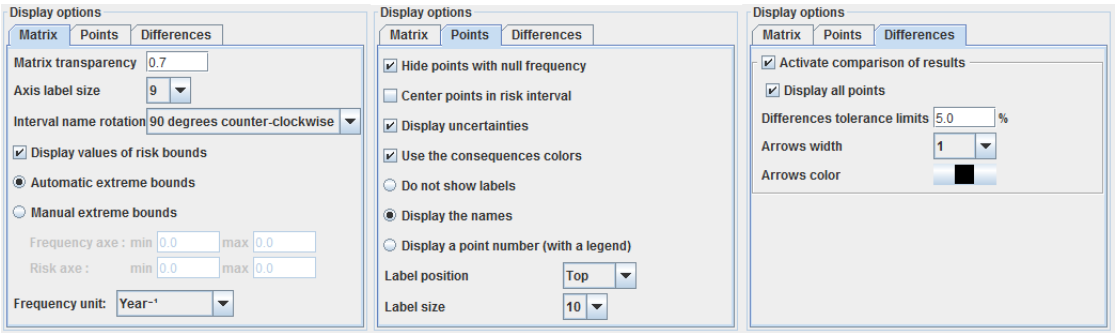
At first, select one or several calculations to be shown:



- **Legende** will be displayed on each points, by prefixing the name of the consequences. It is not necessary to inform one. it enables to identify easyli the origin of the points.
- The **Information** column gives computation results used.

- The **Matrix** column gives the name of the matrix choosen.
- The **Show** column ndicates if user wants to mask or to show this results int the matrix.
- The **Color** column enables to choose the color of points in the matrix.
- the **Style** column enables to choose the form of points.
- **Thickness** column enables to choose the thickness of points.

If several calculations are selected and they concern different matrices, an error message appears.



- In the **Matrix** tab :
  - **Matrix transparency** :Allows to make translucent colors defined on the levels of acceptability. It enables you for example to put in advance the points of the consequences. The input value has to be between 0 (transparent) and 1 (opaque).
  - Option **Hide consequences with null frequency** : Allows to take out of the risk matrix the points which frequency is null.
  - Option **Automatic extremes bounds** : Allows to let the application choose the minimal and maximal extreme bounds of the risk matrix.
  - Option **Manual extreme bounds** : Allows to define the minimal and maximal extreme borders of the risk matrix. Be careful it is not possible to inform values which restrict the intervals defined on the model of the matrix.
  - **Frequency unit** : Allows to modify the unit shown on the frequencies.
- In **Points** tab :
  - **Points style** : Allows to select a predefined geometrical shape for the display of points.
  - **Points size** : Allows to select the diameter of display of points.
  - **Points color** : Allows to select a color for the display of points.
- In the **Labels** tab :
  - Option **Do not show labels consequences** : Allows not to show additional information other than points.
  - Option **Display the name of the consequences** : Allows to show in each points, the name of the of consequence to which it is up.
  - Option **Display the Id of the consequences (with legende)** : Allows to show in each points, the Id of the of consequence to which it is up. A legend is then added to the graph by associating in each id the name of its of consequence.
  - **Label position** : Allows to select a predefined position around the point for the display of the label.
  - **Label size** : Allows to select a predefined police size for the label.
  - **Label color** : Allows to select a color for the label.

13. 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.

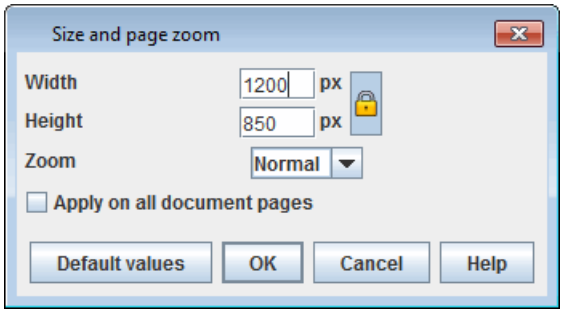
Colour signification is:

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

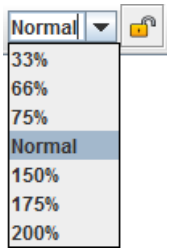
14. Zoom and page size

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.

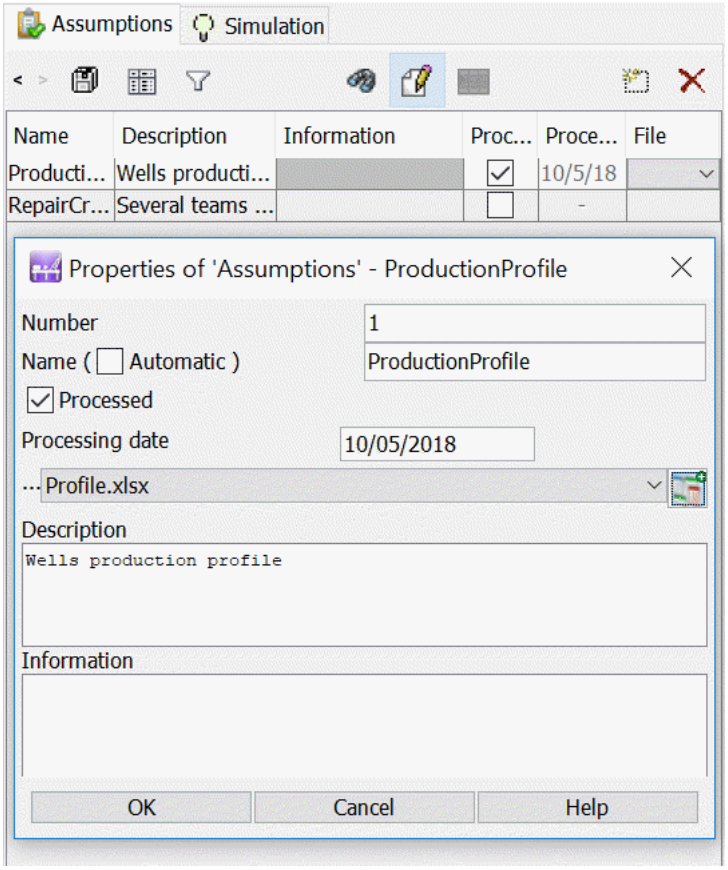
The padlock on the toolbar is used to apply the zoom to the current page or to all pages in the document.

	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.

15. Hypothesis

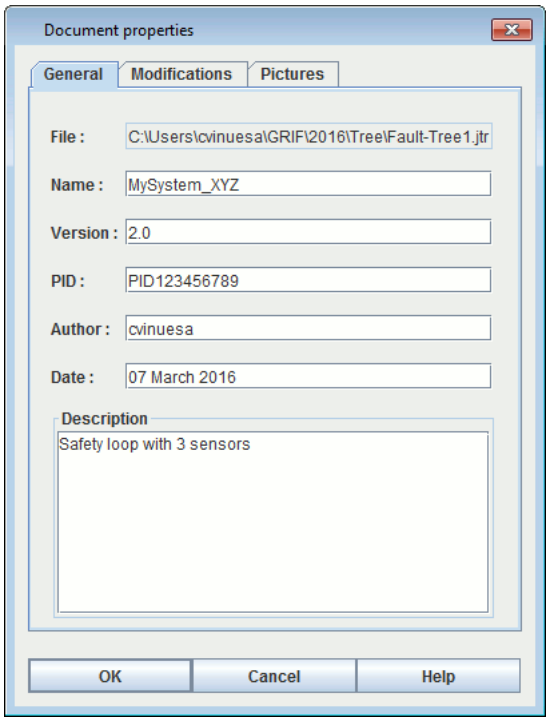
In the data table, in **Hypothesis** tab, it is possible to follow-up and track the studies hypothesis.



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

16. Document properties / Track change / Images management

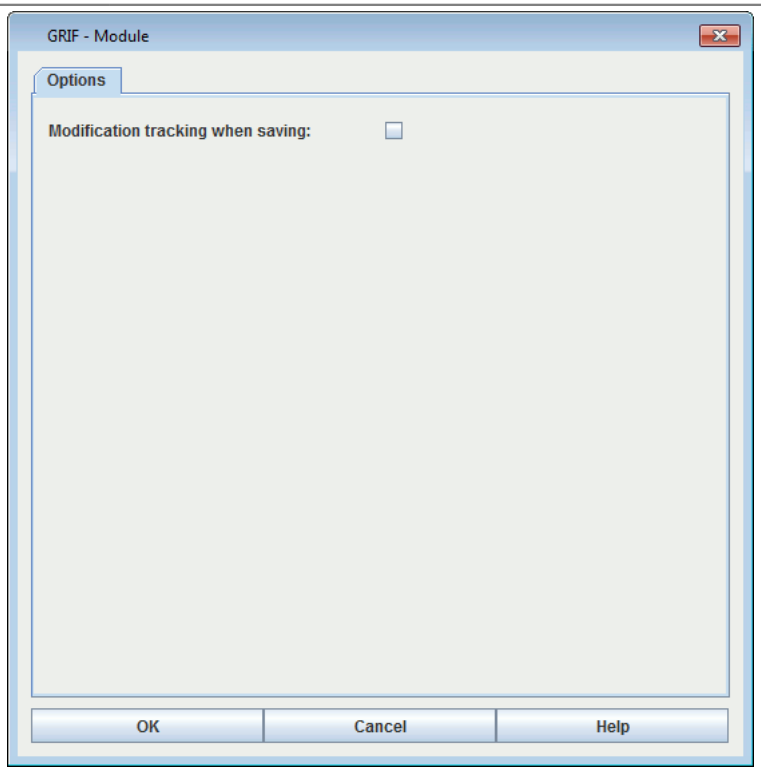
File - Document properties menu enable to save information about document: name, version, comment, ... These information are available in **General** tab.



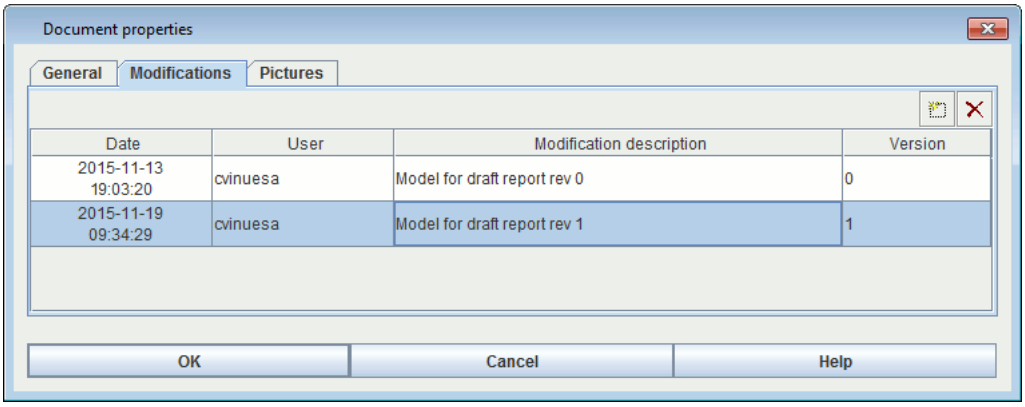
Modification tab enables to save A history of the modifications.

There are two different ways to save modifications:

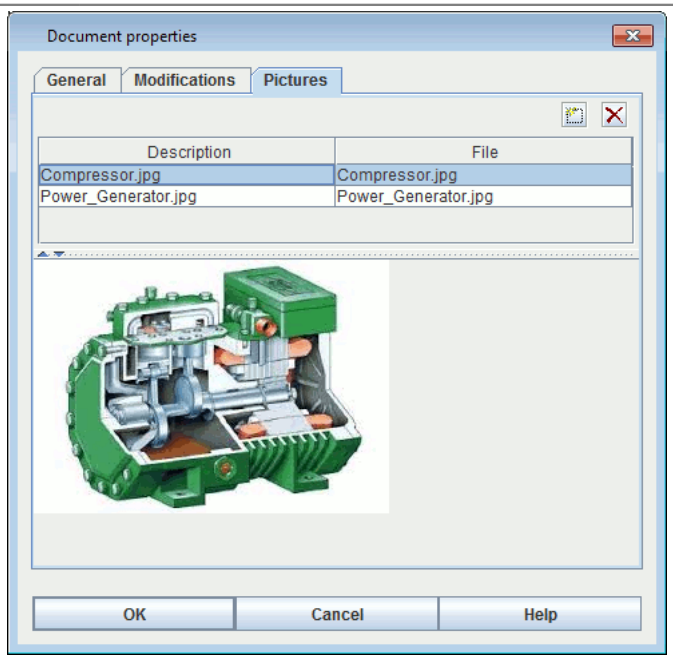
- At each saving by checking: **Modification track when saving** dans **Tools - Document (or Application) options** .



- 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.



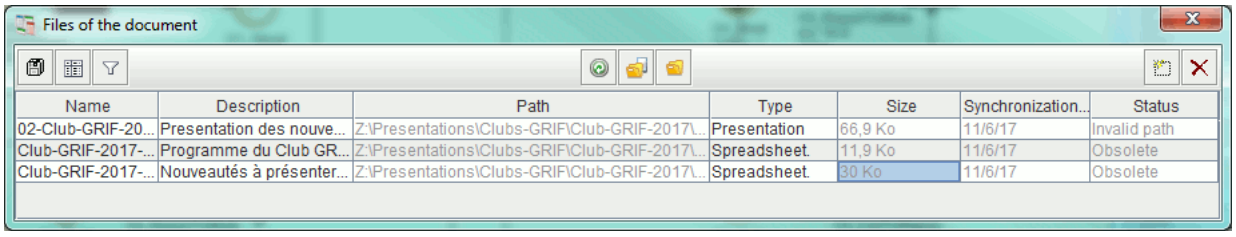
To add a new picture into document, use icon. A double click in **File** column enables to select a picture (jpg, gif or png). A double click in **Description** column enables to give a name or a description to selected image.

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.

17. Files of the documents

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



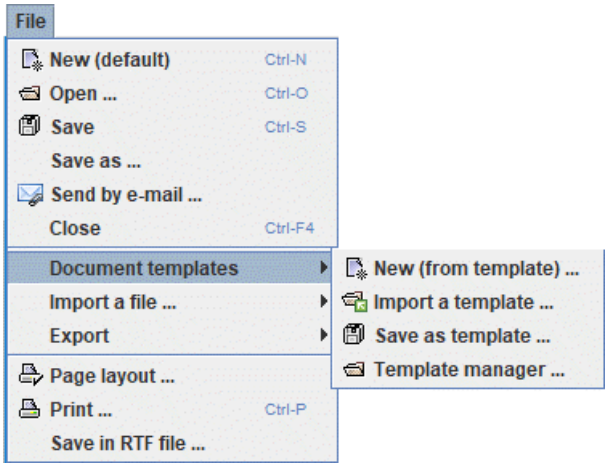
Name	Description	Path	Type	Size	Synchronization...	Status
02-Club-GRIF-20...	Presentation des nouve...	Z:\Presentations\Clubs-GRIF\Club-GRIF-2017\...	Presentation	66,9 Ko	11/6/17	Invalid path
Club-GRIF-2017-...	Programme du Club GR...	Z:\Presentations\Clubs-GRIF\Club-GRIF-2017\...	Spreadsheet.	11,9 Ko	11/6/17	Obsolete
Club-GRIF-2017-...	Nouveautés à présenter...	Z:\Presentations\Clubs-GRIF\Club-GRIF-2017\...	Spreadsheet.	30 Ko	11/6/17	Obsolete

The following icons allow to:

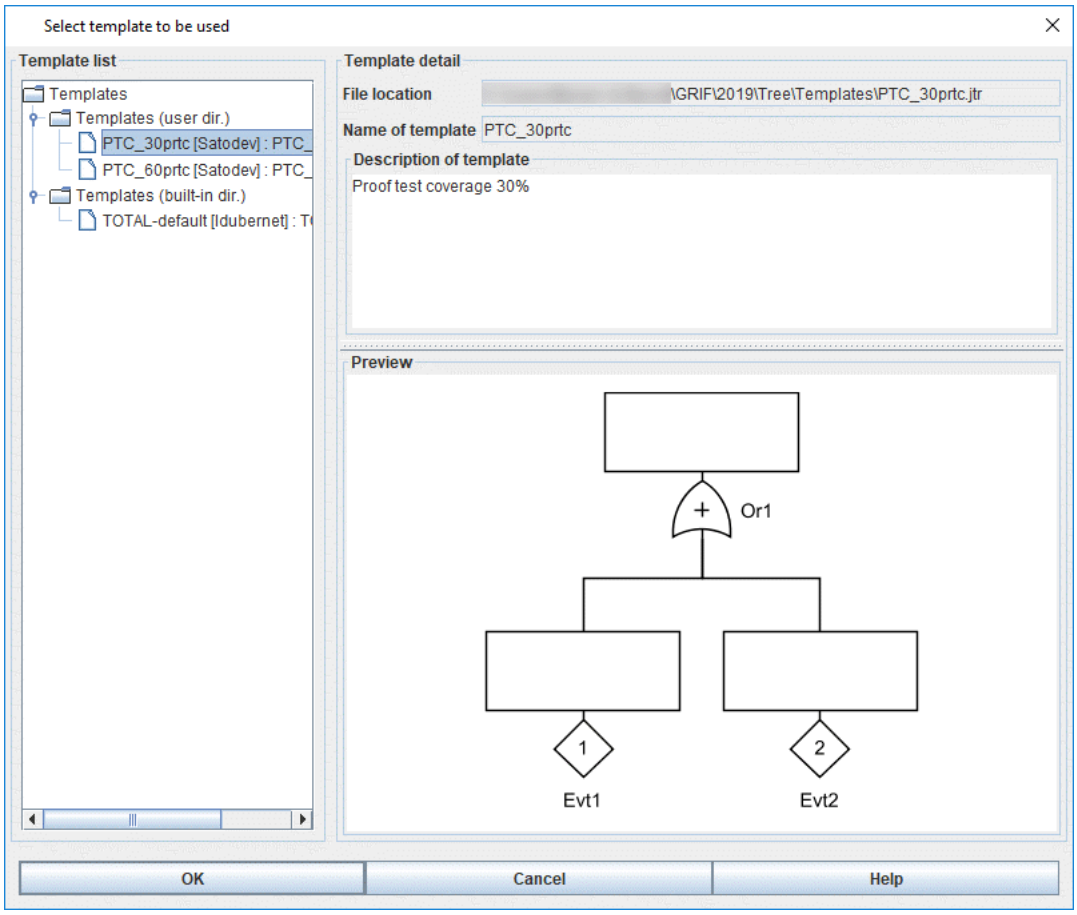
- reload files;
- open files;
- open directory where file is saved.

18. 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.

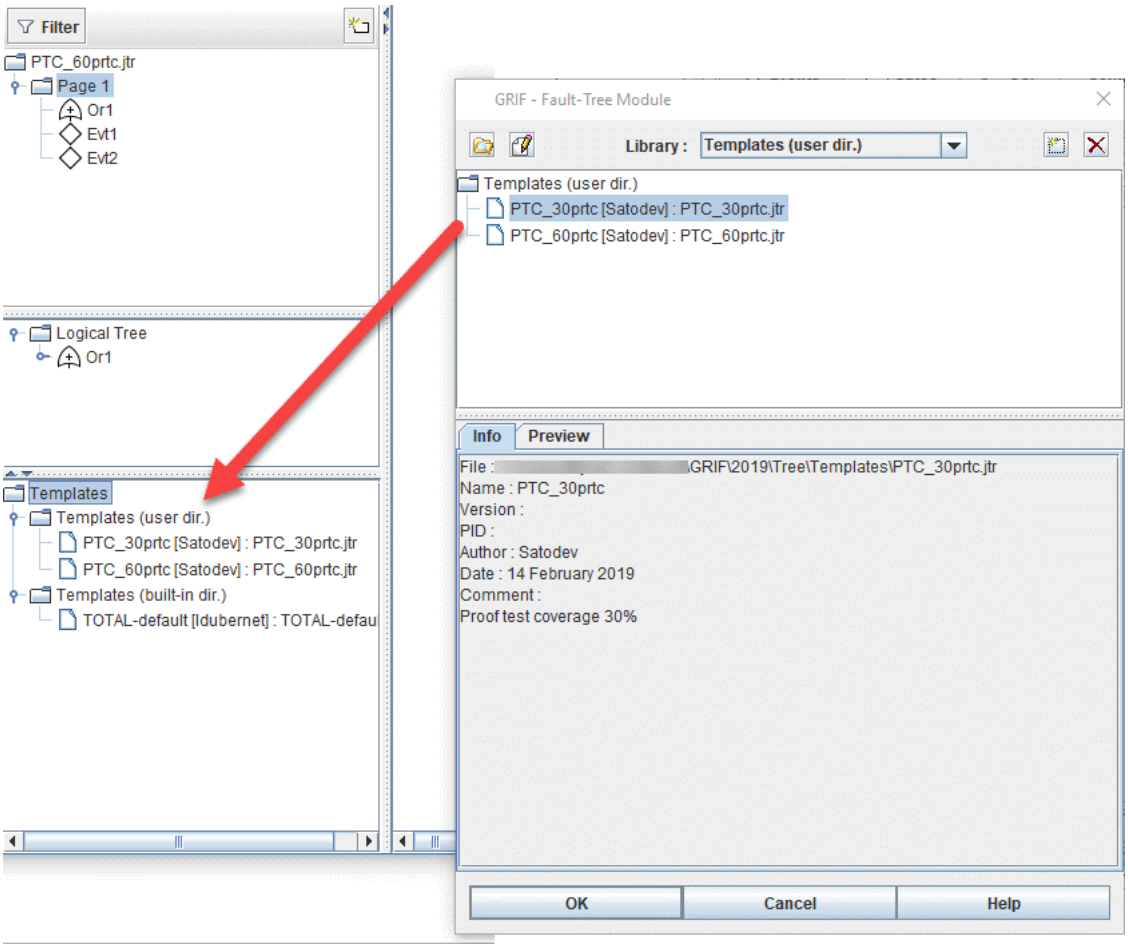


New (from template)... menu enables to open a new document and to initialize it with data from a model already build. A window appears to select the existing model.

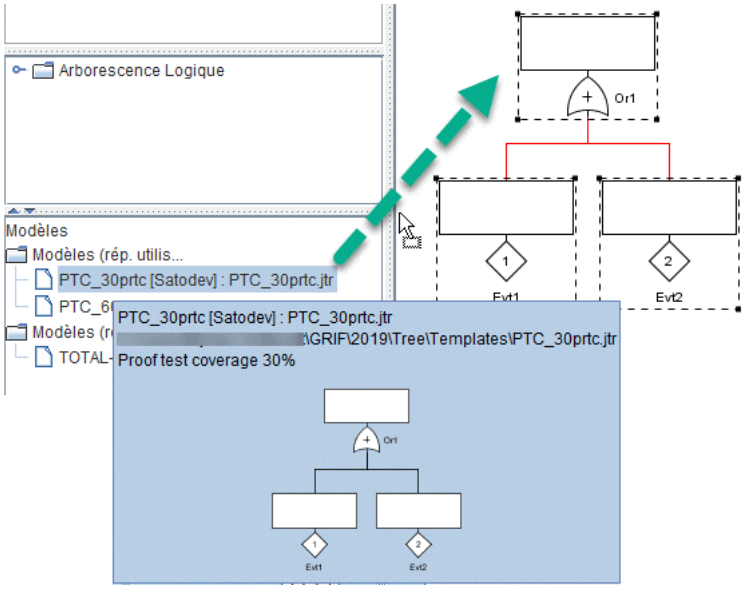


Import a template... menu enables to add to the current document data from a model already build.

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**.

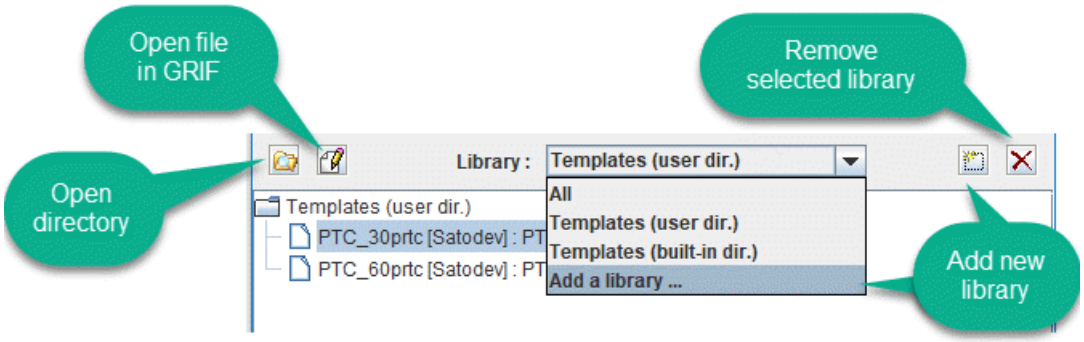


It is possible to create new files from this model using **New (from a template)...** action. A drag and drop to the templates from the input area enables to import the model quickly.



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.

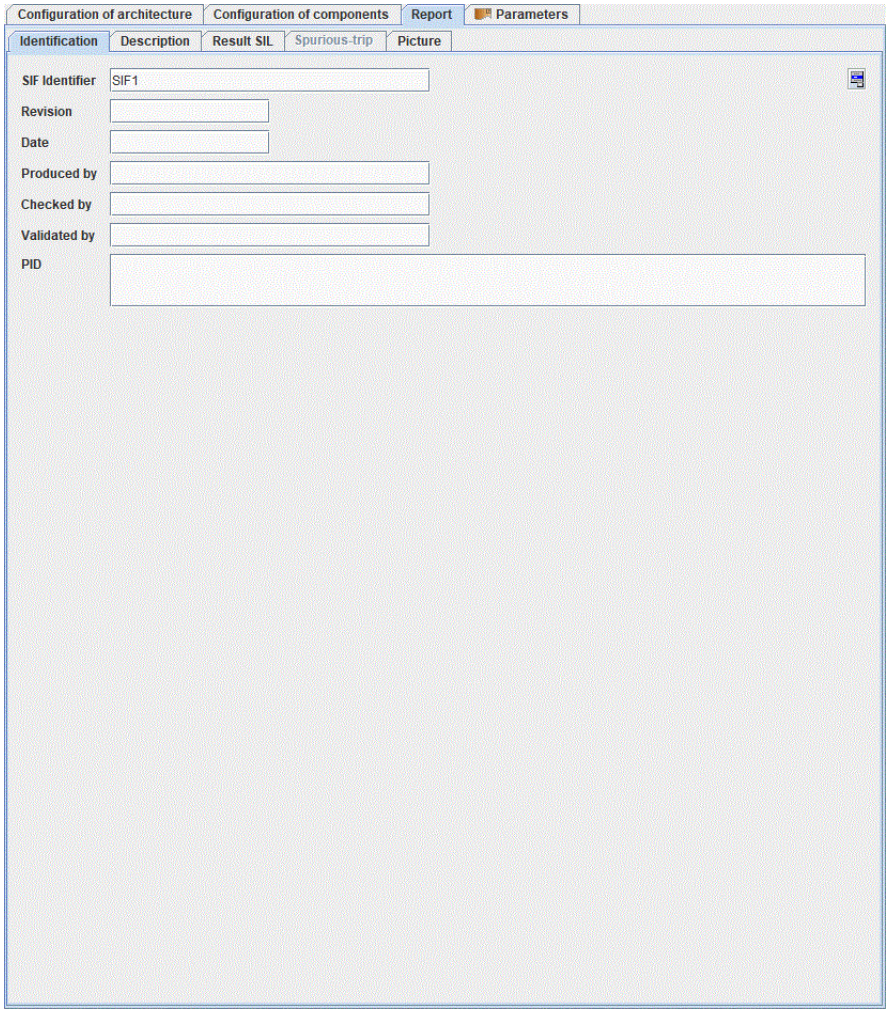
Template manager menu opens a window to manage the template of the document. New document libraries can be added/deleted. To add a new library it is necessary to select a directory of the file system. The tool analyzes the documents in this directory and builds a library that can be used by GRIF based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



19. Generating reports

The PDF reports can be configured in the tab **Report** and its sub-tabs **Identification** and **Description**:

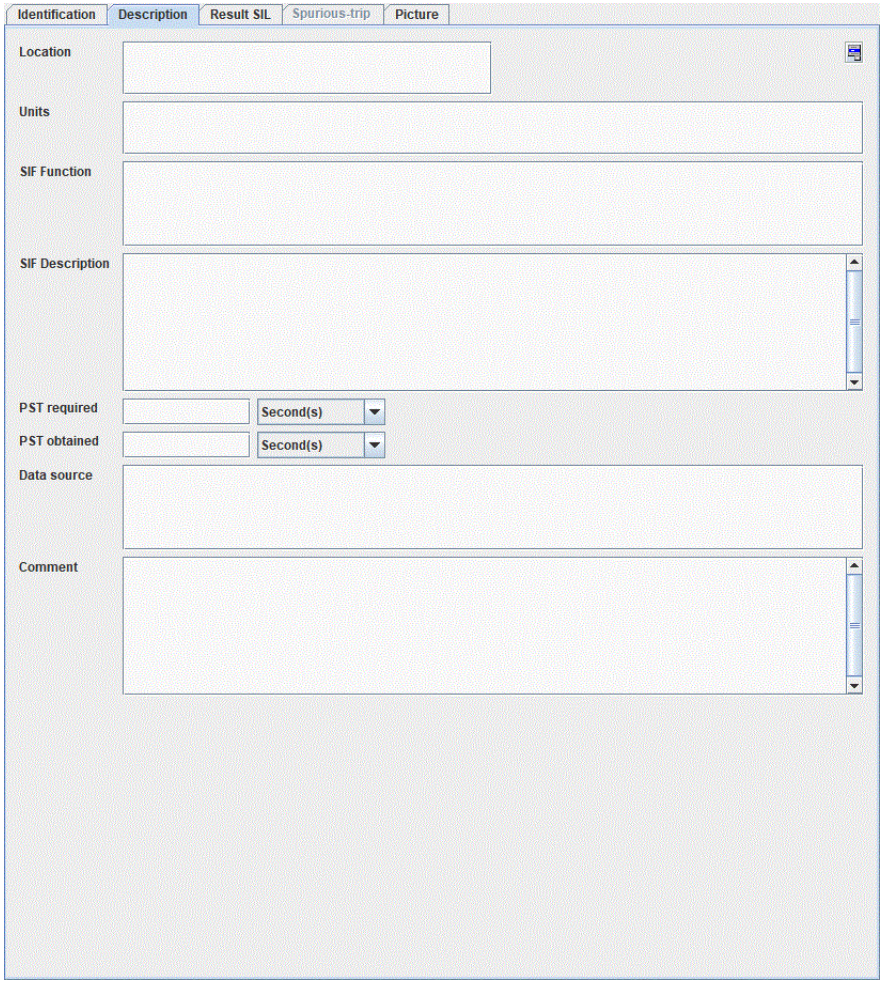
19.1. Identification



When report is generated the following fields are exported:

- **SIF Identifier**: identifier of the SIF or report.
- **Revision**: revision index of the report.
- **Date**: date on which the report was issued.
- **Produced by**: name of the author of the report.
- **Checked by**: name of the checker of the report.
- **Validated by**: name of the person who validated the report
- **PID**: number of the PID.

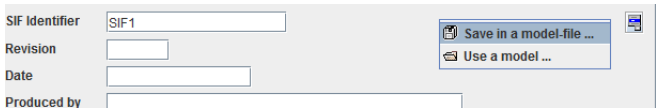
19.2. Description



When report is generated the following fields are exported:

- **Localization**: specify the refinery, the platform, the plant.
- **Units**: specify the units, the sectors, the workshops, the project.
- **SIF Function**: function of the SIF (top event).
- **Description of the SIF**: description of the SIF.
- **PST required**: "Process Safety Time" required with its unit.
- **PST obtained**: "Process Safety Time" obtained with its unit.
- **Data source**: source of the data used in the computations (e.g.: TotalEnergies, EXIDA, OREDA, etc.).
- **Comment**: comments.

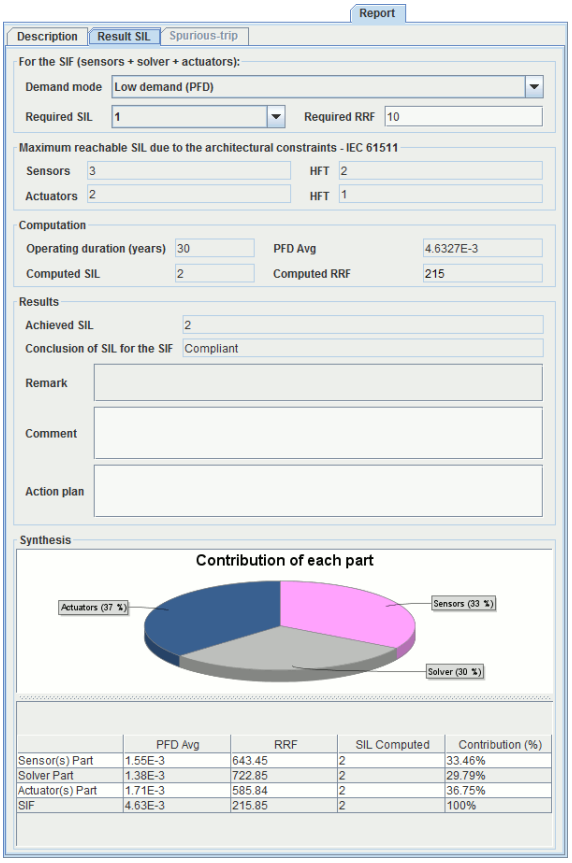
You can save fields of SIF in a model in order to use it for other SIF description. Click on the top right button:



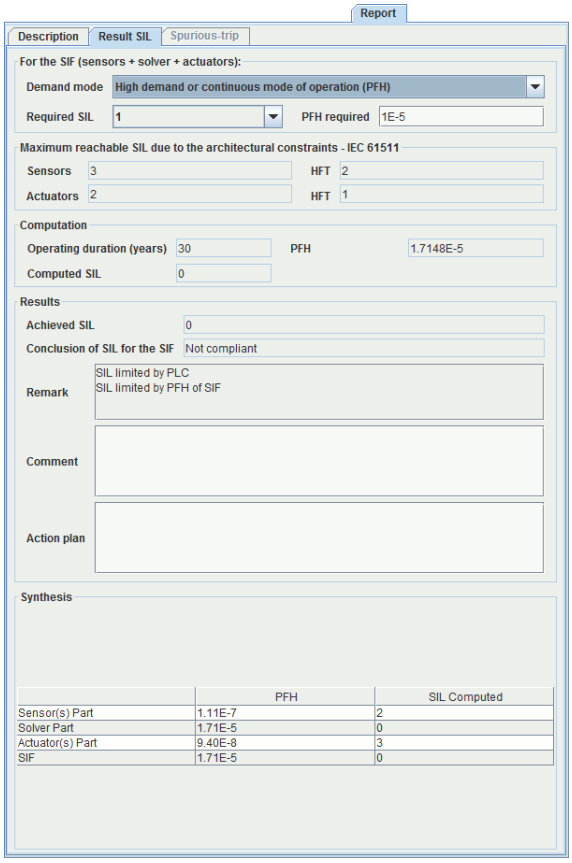
19.3. Result SIL

Result SIL tab do a synthesis of results.

For PFD calculations:

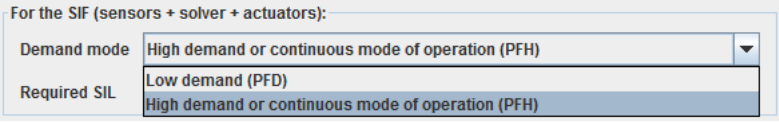


For PFH calculations:



On the top of the tab you must specify the objectives to be reached and sollicitation mode :

- **Demand mode** :



- **Required SIL**: value of the SIL required for the SIF.
- **Required RRF**: value of the RRF required for the SIF.

Then the softwre reminds you the maximum reachable SIL of each part. (not available if many channels)

- **Maximum reachable SIL for sensors**: maximum SIL which can be reached by the sensors due to architectural constraints.
- **Maximum reachable SIL for actuators**: maximum SIL which can be reached by the actuators due to architectural constraints.

The "Computation" part reminds the computed values:

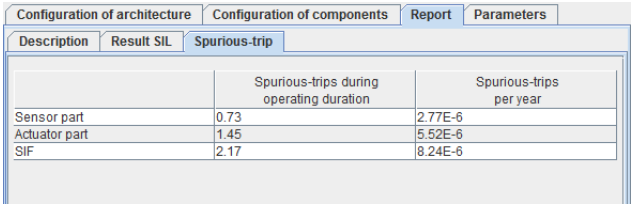
- **Operation duration**: The duration used to do computation.
- **PFD or PFH**: computed PFDAvg or PFH.
- **Computed SIL**: SIL obtained with computed PFD or PFH. Architectural contraints are not taken into account.
- **RRF Calculé**: RRF obtained with computed PFD or PFH.

Then the results part says if objectives are reached or not.

- **Achieved SIL**: SIL obtained for the SIF according to the PFD computation and architectural constraints.
- **Conclusion of SIL for the SIF**: conclusion (compliant or non-compliant).
- **Remark**: Remark generated by the software. It shows the part whose Max-SIL is limiting.
- **Comments**: Comments made by user.
- **Action plan**: List of actions in order, for example, to met the target.

At the end of the tab, a table shows you values for each part in order to identify the most important contributor.

19.4. Spurious-trip



	Spurious-trips during operating duration	Spurious-trips per year
Sensor part	0.73	2.77E-6
Actuator part	1.45	5.52E-6
SIF	2.17	8.24E-6

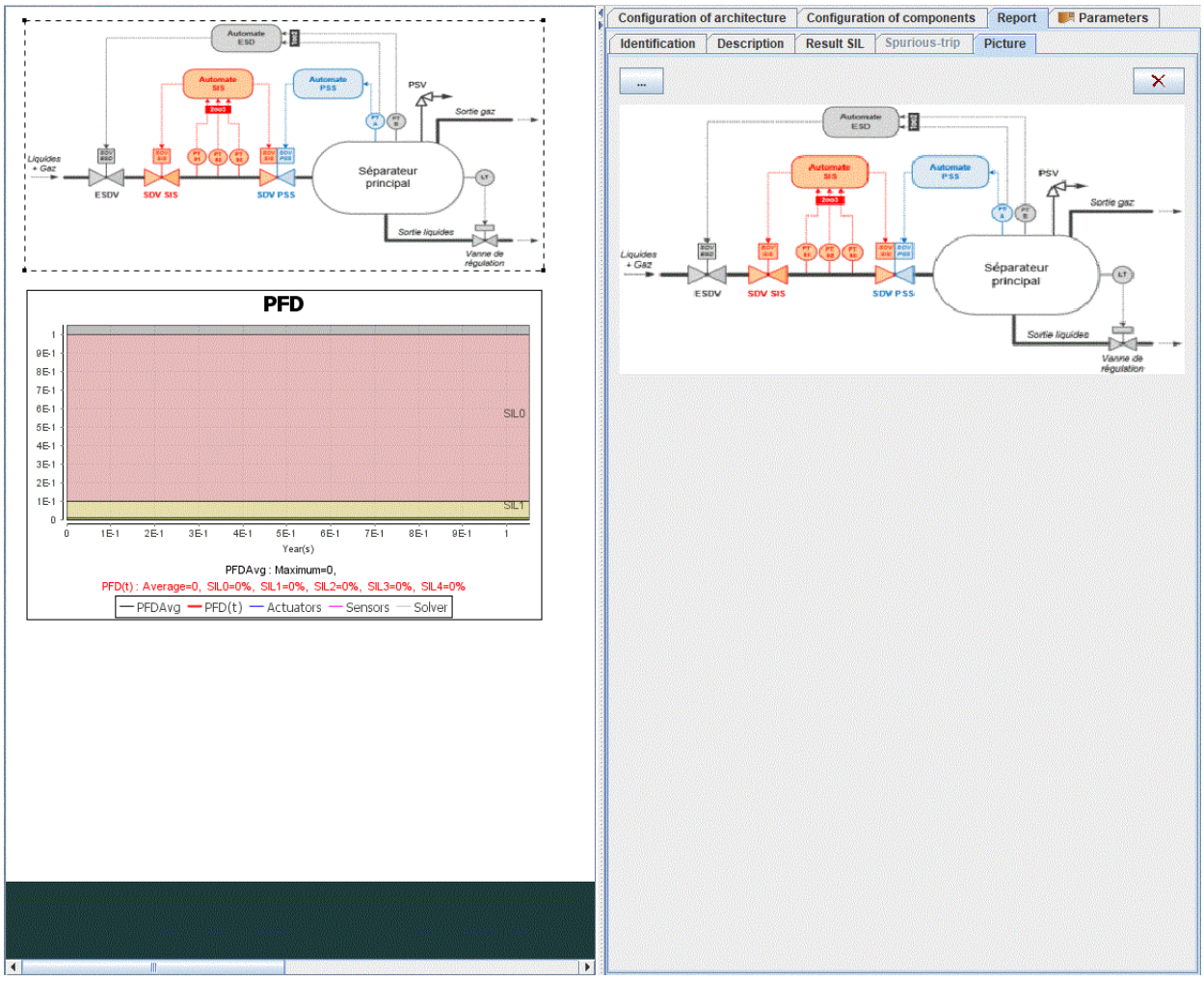
When a sensor has a detected failure, it can lead to the trigger of safety function, even if it is not necessary. It's the same thing for safe failure of actuators, which triggers the safety function. That is why the MooN configuration (with M > 1) are used, the trigger will not be made at the first safe failure.

The SIL module computes the number of expected spurious trips on a given architecture. **Spurious-trip** tab displays computation results of spurious-trips.

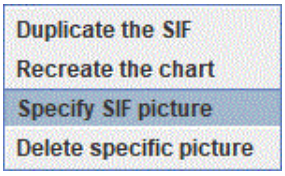
The export of spurious-trip-computation-results in reports is optional. Make sure that the **Spurious-trip in reports** option is selected in Application Options / Options.

19.5. Image

It is possible to assign an image to a SIF, so when the report is generated, the image displayed in the report is this one, instead of the picture of the SIF's architecture. This image can be displayed in the graphical view instead of the graphical representation of the SIF.

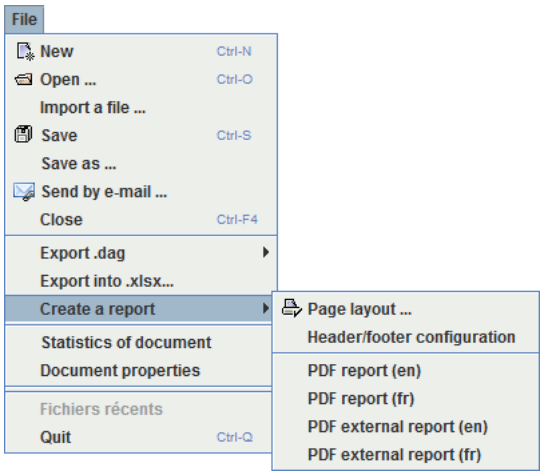


It is possible to modify the image of the SIF directly by clicking on the graphical representation of the SIF, and by clicking on **Specify SIF picture**



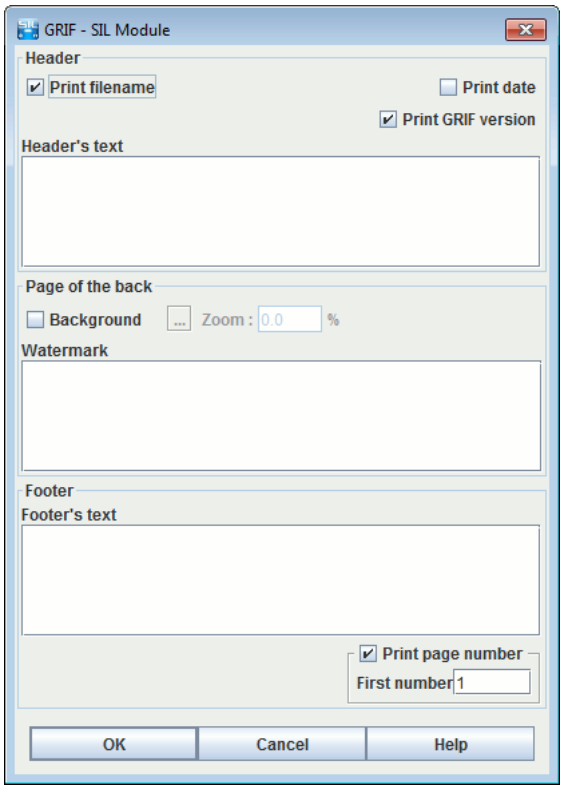
19.6. PDF reports

When all the computations have been carried out, a report can be generated. The language of the report (English or French) can be selected.



The PDF report is generated from the **File/Create a report/PDF report (en)** menu (report written in English) or from the **File/Create a report/PDF report (fr)** menu (report written in French). In all two cases, you must select the location where the PDF file has to be stored and click on save. When the report is generated, it is opened with the programme associated with the PDF format (generally Acrobat Reader).

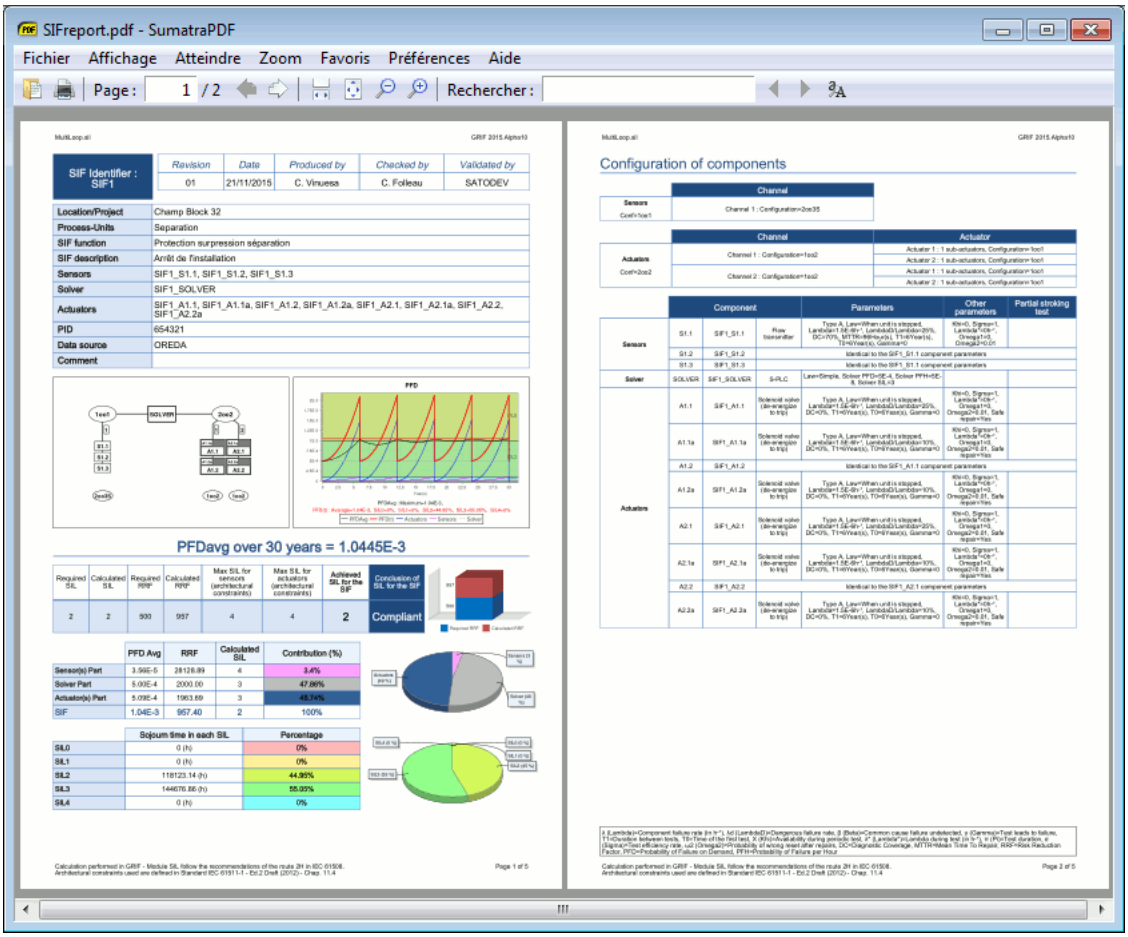
Sub-menu**Header/footer configuration** specify a header, footer and background for the document.



Report is composed with several pages depending on the configuration:

- **Summary (if multiloop calculation)**: synthesis of the different loops with main results and page of detailed results.
- **Results for each SIF**: Results are presented over 2 pages:
  - the first one for results (SIL, risk reduction factor, target met or not, contributions, time spent in each SIL if chacked on **Document options/Export time spent in each SIL**) and SIF references.
  - the second one for configuration of components (type of component, data used, etc...).
- **Test periods synthesis**: test periods for all components are summarized in one table.
- **Actions plan**: if any, they are gathered in the same table.

PDF report:



19.7. Microsoft Excel XLSX file format export

You can also export to XLSX format thanks to **File** menu. The file is made of two tabs, the first one for SIF description and results, the second one for configuration of components.

20. Checksum

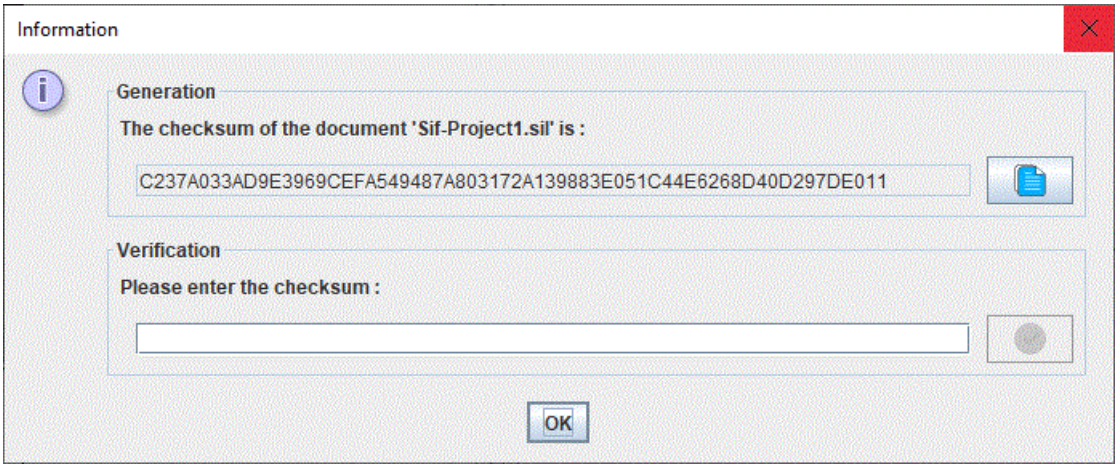
The checksum is a sequence of numbers and letters which enables to identify precisely a document. There can be only one unique checksum by document, and only a simple variation of a property on this document will change this checksum. When someone send you a SIL document, and you have to be sure that this document is identical to the one of the sender (there might sometimes have some loss of datas when a large document is sent with a bad internet connexion), you can ask him this checksum from his document, so you can verify if the checksum is identical.

The creation of this checksum will depend on two parts of your SIL document :

- The **logical** part of your document (Numbers of SIF loops, numbers of sensors, ...)
- The **options** of the document.

The graphical part of the document does not count in the creation of this checksum. Move a graphical component will not change the checksum. The result part of the document, which is created thanks to the logical part and the options of the document, is neither taken into account, to allow the generation of this checksum to be faster.

The actions related to the checksum, like its generation or its verification, are available from the menu **File** -> **Checksum**:



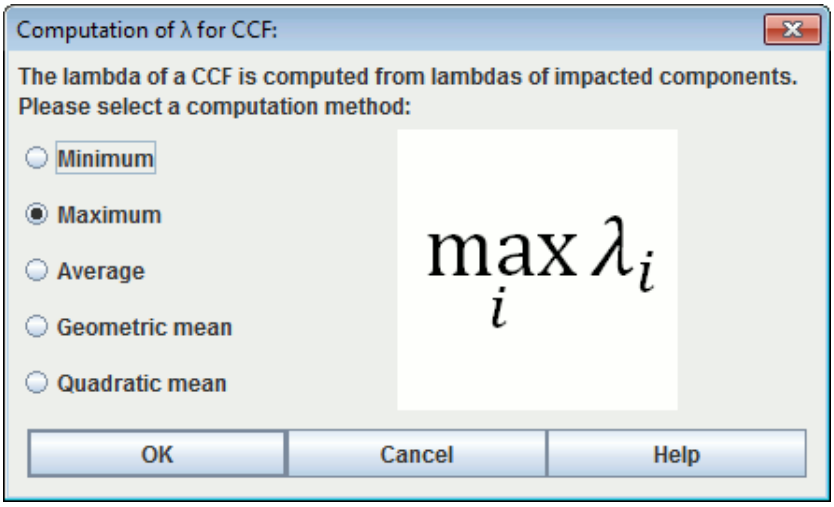
The first part displays the checksum of the current document. A button is located on the right of this checksum, which enables to copy it on the clipboard.

The second part enables to verify if a checksum given by someone is identical to the generated checksum of the document received.

A. Configuration of Lambda computation method for CCF

When using common cause failure, the software must compute a lambdaCCF that will be used for CCF. It is the one that will be multiplied by Beta. Assuming each component impacted bay a CCF has a different lambda, there are many methods to compute the lambdaCCF from the list of lambdas. Five methods are available:

- Minimum: This method uses the minimum value of lambdas. Not recommanded.
- Maximum: Uses the maximum value of lambdas to be concervativ. This method was used in GRIF 2013 and previous version. It can be penalizing when lambda of components are very different.
- Average: This method uses the artimetic mean of lambdas.
- Geometric mean (Method detailed in PDS): This method uses the geometric mean of lambdas. It is PDS Method recommanded by SINTEF. It works fine with very different lambdas.
- Quadratic mean: This method uses the quadratic mean of lambdas.



B. Data Editing Tables

1. Description of the Tables

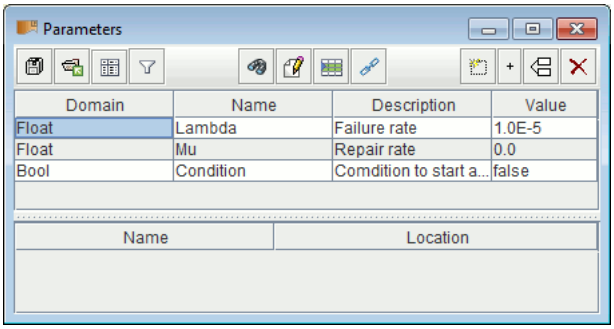
To create or modify data (parameters, variables, etc.), tables are available in the **Data and Computations menu** and in tabs at the right of the view. All the GRIF 2022 data tables operate in the same manner.

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.

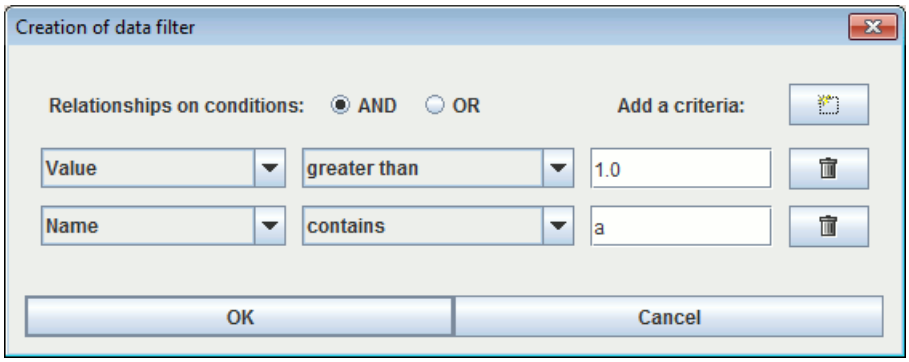
	Saves the table in a text file.
	Import data from another SIL model or from CSV file.
	Opens the column manager (cf. Section 1.2, “Column manager”).
	Displays a panel for searching or filtering data (cf. Section 1.1, “Filter and sorting data”).
	When the **display selection** button is pressed, a click 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.

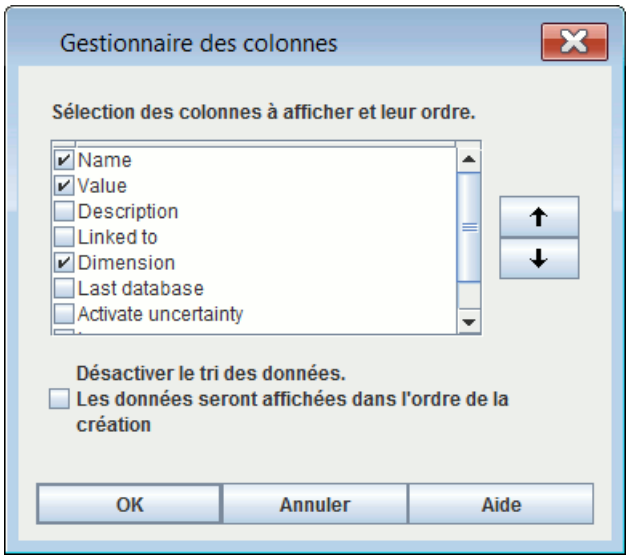
If the Boolean expression is true, the data will be kept (displayed); otherwise the data will be masked. When the filter is enabled its value is displayed between < and >.

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:

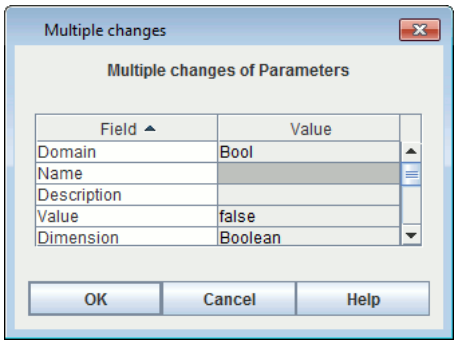


You can choose the columns to be displayed by selecting (or deselecting) the corresponding check boxes. The arrows on the right are used to move the columns up or down in the list to choose the order of the columns. The **Disable data sorting** check box disables the data sorting. This improves the application's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

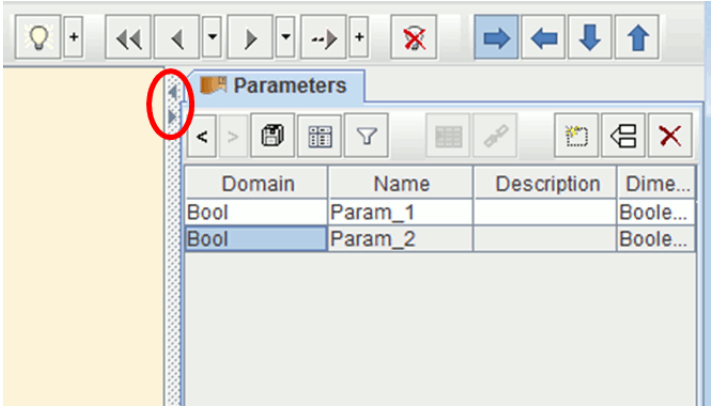


Items which cannot be modified are greyed. The white lines indicate that the selected data does not have the same value for the field in question. A new value can be entered which will be taken into account for all the selected data. The lines with no background colour indicate that all the selected 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.

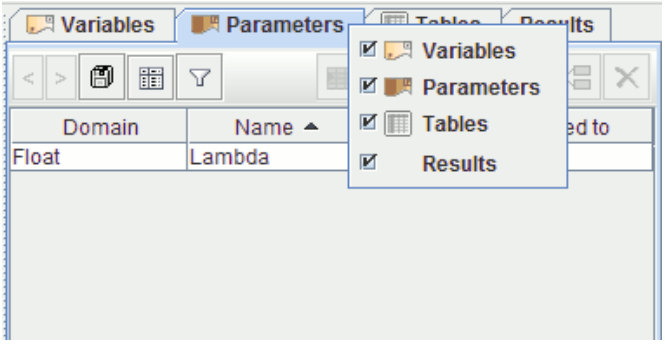
2. Table accessibility

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.


Variables
Parameters
Variables
Parameters
Tables
Results
Domain
Name
Float
Lambda

C. List of components

1. Sensors

- **ANALYSER_PROBE** : Analyser probe
- **CURRENT_TRANSMITTER** : Current transmitter
- **DENSITY_SENSOR** : Density sensor
- **DENSITY_TRANSMITTER** : Density transmitter
- **DETECTOR_BURNER_FLAME** : Burner flame detector
- **DETECTOR_FIRE** : Fire detector
- **DETECTOR_GAS** : Gas detector
- **DETECTOR_PILOT_FLAME** : Pilot flame detector
- **DETECTOR_VIBRATION** : Vibration detector
- **ENERGIZE_TO_TRIP_LINK** : Energize to trip link
- **INPUT_ANALOG** : Analog Input
- **INPUT_CARD** : Input card
- **INPUT_DIGITAL** : Digital Input
- **IS_BARRIER** : IS Barrier
- **MISC_CONVERTER** : Miscellaneous converter
- **MISC_INSTRUMENT** : Miscellaneous instrument
- **MISC_SENSOR** : Miscellaneous sensor
- **MOTOR_STATUS** : Motor status
- **POSITION_SENSOR** : Position sensor
- **POSITION_TRANSMITTER** : Position transmitter
- **POWER_SUPPLY** : Power supply
- **POWER_TRANSMITTER** : Power transmitter
- **PUSH_BUTTON** : Push button
- **RELAY** : Relay
- **RELAY_SAFETY** : Safety Relay
- **ROTATION_CONTROLLER** : Rotation controller
- **SPEED_SENSOR** : Speed sensor
- **SPEED_TRANSMITTER** : Speed transmitter
- **SURGE_PROTECTOR** : Surge protector
- **SWITCH_FIRE** : Fire switch
- **SWITCH_FLOW** : Flow switch
- **SWITCH_GAS** : Gas switch
- **SWITCH_LEVEL** : Level switch
- **SWITCH_LIMIT** : Limit switch
- **SWITCH_PRESSURE** : Pressure switch
- **SWITCH_SPEED** : Speed switch
- **SWITCH_TEMPERATURE** : Temperature switch
- **TEMPERATURE_PROBE** : Temperature probe
- **TORQUE_CONTACTOR** : Torque contactor
- **TORQUE_SENSOR** : Torque sensor
- **TORQUE_TRANSMITTER** : Torque transmitter
- **TRANSMITTER_ANALYZER** : Transmitter/Analyzer
- **TRANSMITTER_FLOW** : Flow transmitter
- **TRANSMITTER_LEVEL** : Level transmitter
- **TRANSMITTER_POSITION** : Position transmitter
- **TRANSMITTER_PRESSURE** : Pressure transmitter
- **TRANSMITTER_SPEED** : Speed transmitter
- **TRANSMITTER_TEMPERATURE** : Temperature transmitter
- **TRIP_AMPLIFIER** : Trip Amplifier
- **VIBRATION_CONTACTOR** : Vibration contactor
- **VIBRATION_SENSOR** : Vibration sensor
- **VIBRATION_TRANSMITTER** : Vibration transmitter
- **VICOSITY_TRANSMITTER** : Viscosity transmitter
- **VISCOSITY_SENSOR** : Viscosity sensor

- **VOLTAGE_TRANSMITTER** : Voltage transmitter
- **WEIGHT_SENSOR** : Weight sensor
- **WEIGHT_TRANSMITTER** : Weight transmitter

2. Actuators

- **CIRCUIT_BREAKER** : Circuit Breaker
- **CONTACTOR** : Contactor
- **HARDWIRED_MOTOR_CONTROL_LOGIC** : Hardwired motor control logic
- **LINK_ENERGIZE_TO_TRIP** : Energize to trip link
- **MISC_ACTUATOR** : Miscellaneous actuator
- **MISC_CONVERTER** : Miscellaneous converter
- **MISC_INSTRUMENT** : Miscellaneous instrument
- **MOTOR_CONTRACTOR** : Motor contactor
- **MOTOR_CONTROL_LOGIC** : Digital motor control logic
- **MOTOR_ELECTRIC** : Electric motor
- **OUPUT_DIGITAL** : Digital Output
- **OUTPUT_CARD** : Output card
- **RELAY** : Relay
- **RELAY_SAFETY** : Safety Relay
- **SUPPLY_PNEUMATIC_HYDRAULIC** : Pneumatic/hydraulic supply
- **SUPPLY_POWER** : Power supply
- **SURGE_PROTECTOR** : Surge protector
- **VALVE** : On/off valve
- **VALVE_AIR** : Air operated on/off valve
- **VALVE_CONTROL** : Control valve with actuator
- **VALVE_DELUGE** : Deluge on/off valve
- **VALVE_DEPRESSURIZATION** : Depressurization on/off valve
- **VALVE_ELECTRICAL** : Electrical on/off valve
- **VALVE_HYDRAULIC** : Hydraulic on/off valve
- **VALVE_SOLENOID_DE_ENERGIZE** : Solenoid valve (de-energize to trip)
- **VALVE_SOLENOID_ENERGIZE** : Solenoid valve (energize to trip)
- **VARIABLE_SPEED_DRIVE** : Variable Speed Drive

3. Solver

- **CERTIFIED_SAFETY_RELAY** : Certified safety relays
- **DCS** : DCS
- **OTHER** : Other
- **PLC** : PLC
- **SOLID_STATE** : Solid State
- **S_PLC** : S-PLC

D. Options of GRIF - SIL

1. Options of GRIF - SIL

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

1.1. 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.
- **Default directory for PDF report** : Default path for the creation of PDF report
- **Default directory for default component** : Default path for the use of the model of component

1.2. 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 Moca version 12 path.

1.3. 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 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.
- **Hide not computed additional info** : Additional information under nodes won't be displayed if it is related to a not computed result. It prevents from multiple "?" display.
- **Display the report picture instead of schema.** : It displays the picture of the SIF that is selected by user in report tab. The default schema won't be displayed.

1.4. Digital format

Digital format tab enables to customize digits display :

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

1.5. 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).
- **Apply modification factor on laws** : Enables probability modification factor. If checked, an "Apply factor" check-box will be available at the bottom of law editing panel.
- **Unit choice for law parameters.** : Activate unit selection for each parameter in law edition windows.
- **Simplify the partial tests on the components** : The partial tests which have a partial test law with a small maximum value will be replaced by a constant law. This will reduce significantly the number of points in the result file, if combined with the option of reduction of points.
- **Use CCF with shock models** : Use this option if your SIF contains a more than 3 redundant sensors/actuators. CCF with shocks model are more realistic than Beta Factor model that is too conservative in these cases.

1.6. Export

Export options: PDF and XLSX :

- **Spurious-trip in reports** : Display the spurious trip rate in XLSX and PDF report.
- **Export summary (PDF)** : Display a summary with the different loops in PDF report.
- **Export test periods of components (PDF)** : Display a summary of the test periods of components in PDF report.
- **Export time spent in each SIL (XLSX, PDF)** : Display a summary of the test periods of components in PDF report.
- **Export a summary of the actions to be made (PDF)** : Display a summary of the actions to be made at the end of PDF report.
- **Export picture of the SIF in report tab (XLSX, PDF)** : Export the picture of the SIF that is selected by user in report tab. The default schema won't be exported.
- **Export attributes of the components in report tab (XLSX, PDF)** : Export attributes of the components (Solver, Actuators, Sensors).
- **Export the files attached to the document (PDF)** : The files attached to the document will be exported as an annex to the PDF

1.7. Curves

Charts tab enables to change charts drawing :

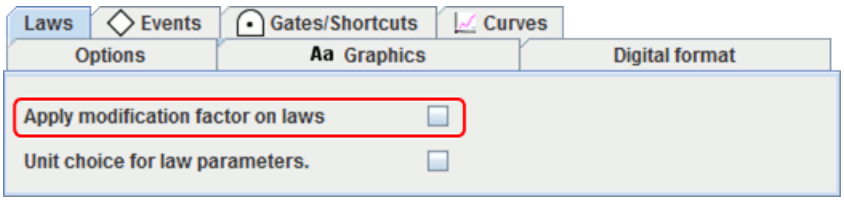
- **Set graphics borders** : Add borders to charts.
- **Set generic values borders** : Add borders to generic values under charts.

- **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.

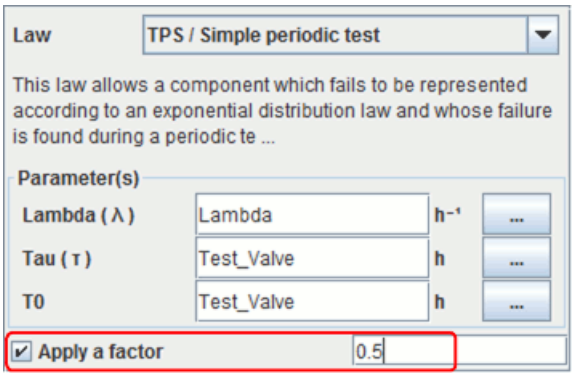
E. Law

1. Description of the laws

A modifier factor can be applied in all the lawx by checking **Apply modification factor on laws** in **document options**.



Once the option selected, a field appears in the events to inform the factor:



In this case, the law is defined by:

$$Q(t) = factor * Qref(t)$$

1.1. UNDEF / Undefined

This law used as default law indicates as user, with an error message in the computation launching, that default law was not changed.

1.2. CST/ Constant law

This law has two parameters: the probability **q** and the inconditional failure rate **w** of the event. Whatever the time, the probability of the component failing is constant.

Parameter:

- **q** (Probability)
- **w** (Inconditional failure rate)

The law is defined as follows:

$$Q(t) = q$$

This law generally corresponds to the case where the only failure considered for the components is that of a refusal to change state (e.g.: Fails to start/stop, etc.).

1.3. EXP / Exponential law

This law only has a one parameter: the component's failure rate (supposed to be constant over time). It describes the time interval before the first failure for a non-repairable component.

Parameters:

- **Lambda** (Rate) = failure rate

The law is defined as follows:

$$Q(t) = 1 - e^{-\lambda t}$$

This law is widely used since it is almost the only one to make it possible to obtain analytical results. In addition, it describes the lifetime of a non-repairable component very well (at least when there are a large number of components) when the component is no longer young.

1.4. EXPD / Dormant exponantial

This law is used to model the dormant events in a more precise way than with a simple dormant law. It has three parameters: the failure rate of thecomponent (supposed constant during the time), the test periodicity and the mission time. This last parameter is not seizes by the user. It corresponds to the last wanted calculation.

Parameters:

- **Lambda** (Rate) = failure rate
- **Tau** (Duration) = test period (time interval between two consecutive tests)
- **Tmax** (Time) = mission time (i.e. t maximum for all t to calculate) This parameter is automaticcally generated according to the last computation time.

The law is defined as follows:

$$Q(t) = 1 - exp^{-\lambda * t} \qquad si\ Tmax \geq \tau$$
$$Q(t) = 1 - exp^{-\lambda * \frac{\tau}{Tmax} * t} \qquad si\ Tmax < \tau$$

Results can be different when step by step calculation is made due to the fact that mission time is the maximum time calculation).

1.5. IND / Unavailability law

This law describes the behaviour of a component (repairable or not), with (or without) failure to start, using exponential expressions. It generalises the exponential law with the **Lambda** parameter (failure rate).

Parameters:

- **Gamma** (Probability) = probability of initial start failure (at t = 0)
- **Lambda** (Rate) = failure rate
- **Mu** (Rate) = repair rate

The law is defined as follows:

$$Q(t) = \frac{\lambda}{\lambda + \mu} - \frac{\lambda - \gamma(\lambda + \mu)}{\lambda + \mu} \times e^{-(\lambda+\mu)t}$$

The **Gamma** and **Mu** parameters are optional. Depending on the case, they can be zero.

- If the component is not repairable, set **Mu** to zero.
- If the component cannot fail to start, set **Gamma** to zero.

The failure to start is only taken into account at t = 0.

1.6. WBL / Weibull

This law has three parameters: **alpha**, **beta** and **t0**. It describes the behaviour of a component which is not repairable and which does not fail to start. Its specific feature is that it takes account of the component's young and old periods.

Parameters:

- **Alpha** (Time) = scale parameter
- **Beta** Factor) = shape parameter
- **T0** (Time) = location parameter

The law is defined as follows:

$$Q(t) = 1 - \exp\left[-\left(\frac{t - t_0}{\alpha}\right)^{\beta}\right]$$

The significance of this law is that new distributions can be tested by varying the **beta** factor:

- If **Beta** is less than 1, the failure rate decreases and the law then allows the period when the component is young to be taken into account.
- If **Beta** is greater than 1, the failure rate increases and the law then allows the component's ageing period to be taken into account.
- If **Beta** is equal to 1, the Weibull law is equivalent to the exponential law.

1.7. WBP / Weibull periodic

This law follows the same logic as the classic Weibull law. It also makes it possible to take into account exclusively periodic preventive maintenance.

Parameters:

- **Age at t=0** (Time) = Virtual age of the component at the initial time.
- **Scale parameter (η)** (Time) = Failure rate scale parameter
- **Shape parameter (β)** (Factor) = Failure rate shape parameter
- **T0** (Time) = First date of preventive maintenance
- **Maintenance period (T1)** (Duration) = Duration between two preventive maintenance
- **Efficiency (α)** (Factor) = Preventive maintenance efficiency (age reduction factor)
- **ARA model** (0 or 1) = Age reduction model:
  - **0** : To use an ARA∞ model Following preventive maintenance (with or without induced corrective maintenance), the age of the component is reduced by a factor α.
  - **1** : To use an ARA1 model Following preventive maintenance (with or without induced corrective maintenance), the age of the component taken since the last preventive maintenance is reduced by a factor α.
- **Coeff. applicable to the failure rate** (Factor) = If x is the coefficient, the scale parameter will be multiplying by x^(-1/beta)

The definition of the law is as follows:

Either **(δ)** = "age reduction" parameter of the failure rate (in time units) calculated according to the specified ARA model.

$$\boldsymbol{h(t) = \frac{\beta}{\eta^{\beta}} \cdot (t - \delta)^{\beta - 1}}$$

General assumptions of the age reduction model:

- No failure is present at the initial time.
- Failures are only detected during preventive maintenance
- All failures are detected at each preventive maintenance
- The first preventive maintenance is carried out at **T0**
- From **T0**, preventive maintenance is carried out periodically, depending on the period **T1**
- The duration of preventive maintenance is negligible.
- If a fault is detected during preventive maintenance, it is repaired immediately (the duration of corrective maintenance induced is negligible).

1.8. WBP10 / Weibull periodic (10 parameters)

Description	This law, like the Weibull law from which it is derived, makes it possible to model the component's young and old periods. It also makes it possible to take into account periodic maintenance with a different rejuvenation model between preventive and curative maintenance.
Parameters	• **Age at t=0** (AgeV0) (Time) = Virtual age of the component at the initial time. • **Scale parameter (η)** (Time) = Failure rate scale parameter • **Shape parameter (β)** (Factor) = Failure rate shape parameter • **T0** (Time) = First date of preventive maintenance • **Maintenance period (T1)** (Duration) = Duration between two preventive maintenance • **Ara model of preventive maintenance (Mp)** (0 or 1) = Age reduction model of preventive maintenance: – **0** : To use an ARA∞ model. Following preventive maintenance, the age of the component is reduced by a factor αp. – **1** : To use an ARA1 model Following preventive maintenance, the age of the component taken since the last preventive maintenance is reduced by a factor αp. • **Efficiency of preventive maintenance (αp)** (Factor) = Preventive maintenance efficiency (age reduction factor) • **Ara model of corrective maintenance (Mc)** (0 or 1) = Age reduction model of corrective maintenance: – **0** : To use an ARA∞. Following corrective maintenance, the age of the component is reduced by a factor αc. – **1** : To use an ARA1 model. Following corrective maintenance, the age of the component taken since the last preventive maintenance is reduced by a factor αc. • **Efficiency of corrective maintenance(αc)** (Factor) = Corrective maintenance efficiency (age reduction factor) • **Coeff. applicable to the failure rate** (Factor) = If x is the coefficient, the scale parameter will be multiplying by x^(-1/beta)
Definition	*n* = number of preventive maintenances carried out before time t. if t < T0, n = 0 if t >= T0, n = 1 + integer part of (t - T0) / T1 Age(t) = component age reduction function. This value is calculated according to the formulas described in the following subsections.
h(*t*)	$h(t) = \frac{\beta}{\eta^{\beta}} \cdot Age(t)^{\beta-1}$
Q(*t*)	$si\ t < T_0\ :\ Q(t) = 1 - e^{-\int_0^t h(u)du}$ $si\ t \geq T_0\ :\ Q(t) = 1 - e^{-\int_{T_0+(n-1)T_1}^{t} h(u)du}$

$w(t)$	$w(t) = h(t) \cdot (1 - Q(t))$

Text syntax	`'Weibull-periodic' '(' [expr]AgeV0 ',' [expr]eta ',' [expr]beta ',' [expr]t0 ',' [expr]t1 ',' [expr]mp ',' [expr]alphap ',' [expr]mc ',' [expr]alphac ',' [expr]lambdaCoeff ',' time ')'`
	`|`
	`'Weibull-periodic' [expr]AgeV0 [expr]eta [expr]beta [expr]t0 [expr]t1 [expr]mp [expr]alphap [expr]mc [expr]alphac [expr]lambdaCoeff`

XML syntax	`<extern-function name='Weibull-periodic'> [expr]AgeV0 [expr]eta [expr]beta [expr]t0 [expr]t1 [expr]mp [expr]alphap [expr]mc [expr]alphac [expr]lambdaCoeff time </extern-function>`

1.8.1. *Weibull-periodic (10-parameter)* age reduction models

General assumptions	• No failure is present at the initial time. • Failures are only detected during preventive maintenance • All failures are detected at each preventive maintenance • The first preventive maintenance is carried out at **T0** • From **T0**, preventive maintenance is carried out periodically, depending on the period **T1** • If no failure is detected during the preventive maintenance, only the preventive maintenance effectiveness applies (depending on the model selected). • If a failure is detected during preventive maintenance, only the corrective maintenance effectiveness applies (depending on the model selected). • Preventive and corrective maintenance times are negligible.
ARA1 model	As a result of maintenance (preventive or corrective), the age of the component taken since the last preventive maintenance is reduced by a factor of α. If a failure is detected during preventive maintenance, the model selected for corrective maintenance applies with α = αc, otherwise the model selected for preventive maintenance applies with α = αp. NOTE: for the first preventive maintenance, it is the age of the element taken since t0 which is reduced by a factor α.
ARA∞ model	As a result of maintenance (preventive or corrective), the age of the element is reduced by a factor of α. If a failure is detected during preventive maintenance, the model selected for corrective maintenance applies with α = αc, otherwise the model selected for preventive maintenance applies with α = αp. NOTE: the model selected for preventive maintenance may be different from the model selected for corrective maintenance (Mc ≠ Mp).

1.8.2. *Weibull-periodic (10-parameter)* modeling algorithm

t = 0	n = 0 $Age^* = Age_0$
$0 \le t < T_0$	$Age(t) = Age^* + t$
$t = T_0$	n = n + 1 $Age^* = Q(T_0) \cdot [Age^* \cdot (1 - \alpha_c \cdot (1 - M_c)) + T_0 \cdot (1 - \alpha_c)]$

	$+ \ (1-Q(T_0))\cdot[Age^*\cdot(1-\alpha_p\cdot(1-M_p))+T_0\cdot(1-\alpha_p)]$
Begin loop	
$T_0+(n-1)T_1 \leq t < T_0+nT_1$	$Age(t)=Age^*+t-(T_0+(n-1)\cdot T_1)$
$t = T_0 + nT_1$	$n = n + 1$ $Age^*=Q(T_0+n\cdot T_1)\cdot[Age^*\cdot(1-\alpha_c\cdot(1-M_c))+T_1\cdot(1-\alpha_c)]$ $+ \ (1-Q(T_0+n\cdot T_1))\cdot[Age^*\cdot(1-\alpha_p\cdot(1-M_p))+T_1\cdot(1-\alpha_p)]$
Return to beginning of loop	

1.9. WBD / Weibull with detected failures

Model whose failure follows a classical Weibull law and whose repair begins as soon as the failure appears and follows an exponential law with parameter Mu.

Parameters:

- **Age at t=0** (Time) = Virtual age of the component at the initial time.
- **Scale parameter (η)** (Time) = Failure rate scale parameter
- **Shape parameter (β)** (Factor) = Failure rate shape parameter
- **Mu** = Repair rate
- **Coefficient applicable to the failure rate** (Factor) = Given a coefficient x, multiply the scale parameter by x ^ (- 1 / beta))

General assumptions :

- No failure is present at the initial time.
- All faults are detected online (i.e. immediately).
- Repairs begin as soon as faults appear.
- Repairs cause downtime.
- There are no other causes of downtime than repairs.
- The repairs have no effect on the age of the element.

The definition of the law is as follows:

$$h(t) = \frac{\beta}{\eta^{\beta}} \cdot (t + AgeV_0)^{\beta-1}$$

$$Q(t) = 1 - e^{-\left(\frac{t+AgeV_0}{\eta}\right)^{\beta} - \mu.t} \cdot \left[\mu \cdot \left(\int_0^t e^{-\left(\frac{x+AgeV_0}{\eta}\right)^{\beta} + \mu.x} dx\right) + e^{\left(\frac{AgeV_0}{\eta}\right)^{\beta}}\right]$$

1.10. TPS / Simple Periodic Test law

This law allows a component which fails to be represented according to an exponential distribution law and whose failure is found during a periodic test. The repair is then carried out instantaneously.

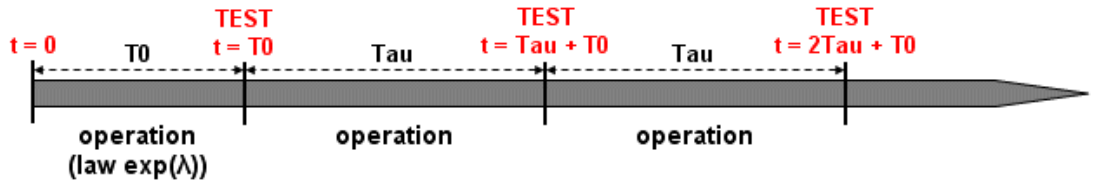
Parameters:

- **Lambda** (Rate) = failure rate
- **Tau** (Duration) = test period (time interval between two consecutive tests)
- **T0** (Time) = date of first test

The law is defined as follows:

$$Q(t) = \begin{cases} 1 - e^{-\lambda t} & if \quad t < t_0 \\ 1 - e^{-\lambda\left[(t-t_0) \bmod \tau\right]} & otherwise \end{cases}$$

Here is a small graph representing the different phases of the component's "life":



This law is a simplified version of the "TPC / Full Periodic Test" law.

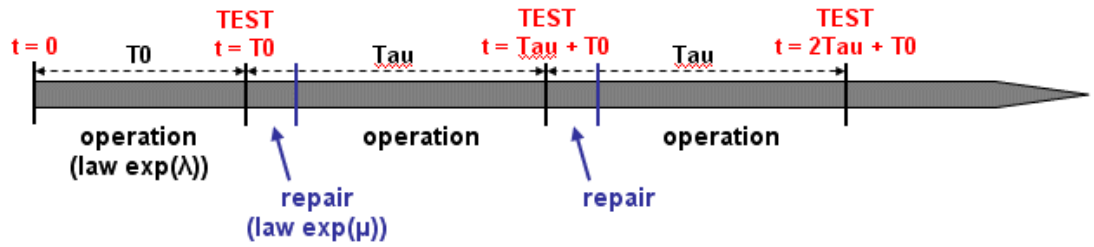
1.11. TPE / Extended Periodic Test law

This law allows a component which fails to be represented according to an exponential distribution law and whose failure is found during a periodic test. The repair phase is then modelled by an exponential of the **Mu** parameter.

Parameters:

- **Lambda** (Rate) = failure rate
- **Mu** (Rate) = repair rate (when the failure has been found during a test)
- **Tau** (Duration) = test period (time interval between two consecutive tests)
- **T0** (Time) = date of first test

Here is a small graph representing the different phases of the component's "life":



This law is a simplified version of the "TPC / Full Periodic Test" law.

1.12. TPC / Full Periodic Test law

This law allows a periodically tested component to be represented as completely as possible. There are many parameters in play.

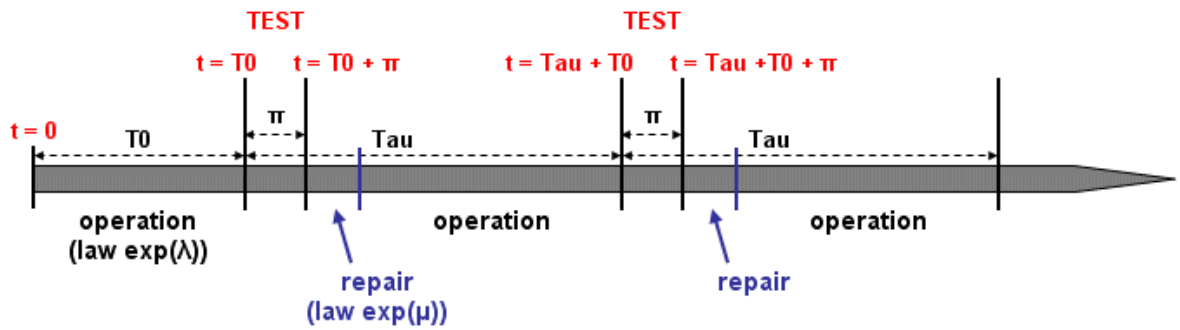
Parameters:

- **Lambda** (Rate) = failure rate during operation or on standby
- **Lambda*** (Rate) = failure rate during the test
- **Mu** (Rate) = repair rate (once the test has shown up the failure)
- **Tau** (Duration) = test period (time interval between two consecutive tests)
- **Theta** (Time) = date of first test (ignore parameter value: **Tau**)

- **Gamma** (Probability) = probability of failure due to starting the test (ignore parameter value: 0 = starting the test does not cause a failure)
- **Pi** (Duration) = duration of test (ignore parameter value: 0 (instantaneous test))
- **X** = (Boolean) indicator of component availability during the test (0 = component unavailable during the test; 1 = component available) (ignore parameter value: 1 = available during the test)
- **Sigma** (Probability) = test cover rate (probability that the component failure is detected during the test) (ignore parameter value: 1 = the test covers all the possible failures)
- **Omega 1** ((Probability) = probability of forgetting to reconfigure after the test (ignore parameter value: 0 = no reconfiguration problem)
- **Omega 2** ((Probability) = probability of forgetting to reconfigure after the repairing (ignore parameter value: 0 = no reconfiguration problem)

the "ignore parameter value" is the value to type if you want parameter to do not affect component availability.

Here is a small graph representing the different phases of the component's "life":



1.13. TPC / Full Periodic Test with defined times

This law is the same as the Full Periodic Test law with 11 parameters (see above). The difference is in times of tests. This law does not have **Tau** or **Teta**, but there is a **Times of tests** parameter where you can specify the times the tests will be made.

1.14. NRD / No Recovery Before Delay law

This law takes two parameters: a repair rate **Mu** and a delay **Delay**. For non repairable components, it gives the probability of not succeeding to recover the component before a delay Delay.

This law does not depend on the time, it is a short version of a constant law.

Parameters:

- **Mu** (Rate) = repair rate
- **d** (Duration) = recovery time

The law is defined as follows:

$$Q(t) = e^{-\mu d}$$

1.15. GLM / GLM Asymptotic law

This law is a variation of the "IND / Unavailability" law. It corresponds to the probability of a "IND / Unavailability" law computed at t = infinity.

This law does not depend on the time, it is a short version of a constant law.

Parameters:

- **Lambda** (Rate) = failure rate
- **Mu** (Rate) = repair rate

The law is defined as follows:

$$Q(t) = \frac{\lambda}{\lambda + \mu}$$

1.16. DOR / Dormant

This law has three parameters: a failure rate, a mean repair time and a delay. In addition, it does not depend on the time.

Parameters:

- **Lambda** (Rate) = failure rate
- **MTTR** (Duration) = average repair time
- **d** (Duration) = delay

The law is defined as follows:

$$Q(t) = \frac{\lambda d - (1 - e^{-\lambda d}) + \lambda . MTTR . (1 - e^{-\lambda d})}{\lambda d + \lambda . MTTR . (1 - e^{-\lambda d})}$$

1.17. CMT / Constant mission time

This law is a simplified case of the "IND / Unavailability" law. It corresponds to an exponential law with a fixed time given as parameter.

This law does not depend on the time, it is a short version of a constant law.

The parameter Q is optional.

Parameters:

- **Lambda** (Rate) = failure rate
- **T** (Duration) = mission time
- **Q** (Probability) = optional law

The law is defined as follows:

$$Q(t) = Q + 1 - e^{-\lambda T}$$

1.18. EMP / Empiric

This not actually en law, you must enter probability and failure rate in a tableau according to the time.

If you ask for computation a times which are not in the table, the value will be interpolated according to other points.

1.19. MKV / Markov model

This law uses a Markov graph as definition. Select the path of the .jma file. In order to do Boolean computation, you need to do a preprocessing of the law. The preprocessing automatically start Markov module and retrieve necessary values. It can be done with a right-click on the object having the law, or in **Data and computations** menu.

1.20. MKVM / Markov matrix

Description

This law allows the use of a monophase Markov graph defined according to its transition matrix. Its use does not require precalculations.

This matrix is stochastic :

- $\forall(i,j)\ P_{ij} >= 0$
- $\forall i\ \sum_j P_{ij} = 1$

Parameters

Number of states (n) : Number of matrix states
Transition matrix : n^2-size vector of P_{ij}, the probability of moving from i to j
Probability at t=0 : n-size vector of probabilities at t=0 for each state
Availablity : n-size vector of availabilities for each state (0=unavailable, 1=available)

Example

Consider the following transition matrix:

State 1	State 2	State 3	State 4	State 5	State 6
-	2.1E-5	0	0	0	0
0	-	1.8E-5	0	0	0
0	0	-	1.5E-5	0	0
0	0	0	-	1.2E-5	0
0	0	0	0	-	9E-6
0	0	0	0	0	-

The following probabilities at t=0:

State 1	State 2	State 3	State 4	State 5	State 6
1	0	0	0	0	0

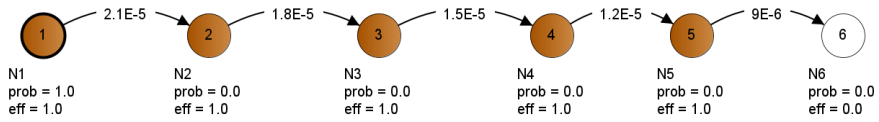
The following availabilities:

State 1	State 2	State 3	State 4	State 5	State 6
1	1	1	1	1	0

The textual syntax to use will be:

```
markov-matrix(time(),6,
        0,2.1E-5,0,0,0,0,
        0,0,1.8E-5,0,0,0,
        0,0,0,1.5E-5,0,0,
        0,0,0,0,1.2E-5,0,
        0,0,0,0,0,9E-6,
        0,0,0,0,0,0,
        1,0.0,0.0,0.0,0.0,0.0,
        1,1,1,1,1,0.0)
```

The equivalent markov graph that will be generated will have the following form:

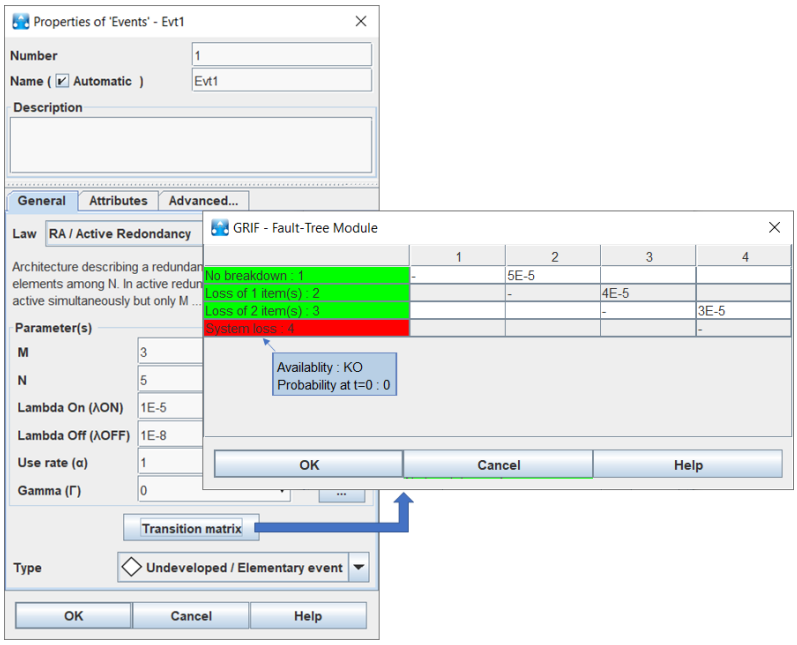


Textual syntax	`'markov-matrix' '(' time ',' [expr]nbStates ','`
	`[expr ',' ... ',' expr]Pij ','`
	`[expr ',' ... ',' expr]init ','`
	`[expr ',' ... ',' expr]avail ')'`

XML syntax	`<extern-function name='markov-matrix'>`
	`time`
	`[expr]nbStates`
	`[expr ... expr]Pij`
	`[expr ... expr]init`
	`[expr ... expr]avail`
	`</extern-function>`

1.21. Redundancy laws

GRIF 2022 offers several functions to calculate the reliability and the availability of a set of elements in redundancy m among n. These functions generate a single-phase Markov graph to perform the calculations. The generated transition matrix is accessible using the **Transition matrix** button displayed below the parameter entry.



The available states are shown in green, the unavailable states in red. By tooltip on the states, we can also display the initial probability of each state.

1.21.1. RA / Active Redundancy

Description	*Albizia* offers several functions to calculate the reliability and the availability of a set of elements in redundancy m among n. These functions generate a single-phase Markov graph to perform the calculations (cf. Section 1.20, “MKVM / Markov matrix”). In active redundancy configuration, the n elements are active simultaneously but only m elements are necessary to ensure the mission.
Parameters	*M* : Number of functional elements required to perform the function, *N* : Total number of items available, *Lambda On (λON)* : Element failure rate when the equipment is turned on, *Lambda Off (λOFF)* : Failure rate of an element when the equipment is switched off, *Alpha (α)* : The use rate α corresponds to the operating time of equipment over the total time of the mission. *Gamma (Γ)* : Probability of failure on demand.

Definitions

$\lambda_{Active} = \alpha * \lambda_{ON} + (1- \alpha) * \lambda_{OFF}$

At t = 0 the probability of being in the nominal state is 1- Γ

At t = 0 the probability of being in the failure state (KO) is Γ

The "System KO" state is a state where the system is unavailable. The system is available in other states.

Transition matrix

	N in operation 0 in fault	**N-1 in operation 1 in fault**	**N-2 in operation 2 in fault**	**...**	**M+1 in operation N-M-1 in fault**	**M in operation N-M in fault**	**M-1 in operation N-M+1 in fault System KO**
N in operation 0 in fault	-	$N * \lambda_{Active}$					
N-1 in operation 1 in fault		-	$(N - 1) * \lambda_{Active}$				
N-2 in operation 2 in fault			-	...			
...				-	$(M + 2) * \lambda_{Active}$		
M+1 in operation N-M-1 in fault					-	$(M + 1) * \lambda_{Active}$	
M in operation N-M in fault						-	$M * \lambda_{Active}$
M-1 in operation N-M+1 in fault Système KO							-

Textual syntax

```
'markov-ra' '(' time ',' [expr]M ',' [expr]N ',' [expr]λON
',' [expr]λOFF ',' [expr]α ',' [expr]Γ ')'
```

XML syntax

```
<extern-function name='markov-ra'>
  time [expr]M  [expr]N  [expr]λON [expr]λOFF [expr]α [expr]Γ
  </extern-function>
```

1.21.2. RP / Passive Redundancy

Description

Albizia offers several functions to calculate the reliability and the availability of a set of elements in redundancy m among n. These functions generate a single-phase Markov graph to perform the calculations (cf. Section 1.20, “MKVM / Markov matrix”).

In passive redundancy, the M elements necessary to ensure the function are active simultaneously. The (N-M) elements are activated successively following faults.

Parameters

M : Number of functional elements required to perform the function,

N : Total number of items available,

Lambda On (λON) : Element failure rate when the equipment is turned on,

Lambda Off (λOFF) : Failure rate of an element when the equipment is switched off,

Alpha (α) : The use rate α corresponds to the operating time of equipment over the total time of the mission.

Gamma (Γ) : Probability of failure on demand.

Definitions

$\lambda = \lambda_{Active} = \alpha * \lambda_{ON} + (1- \alpha) * \lambda_{OFF}$

$\lambda^* = \lambda_{OFF}$

At t = 0 the probability of being in the nominal state is 1 - Γ

At t = 0 the probability of being in the failure state (KO) is Γ

The "System KO" state is a state where the system is unavailable. The system is available in other states.

Transition matrix

	N in operation 0 in fault	**N-1 in operation 1 in fault**	**N-2 in operation 2 in fault**	**...**	**M in operation N-M in fault**	**M-1 in operation N-M+1 in fault System KO**
N in operation 0 in fault	-	Mλ+(N-M)λ*				
N-1 in operation 1 in fault		-	Mλ+(N-M-1)λ*			
N-2 in operation 2 in fault			-	Mλ+(N-M-2)λ*		
...				-	Mλ+λ*	
M in operation N-M in fault					-	Mλ
M-1 in operation N-M+1 in fault System KO						-

Textual syntax

```
'markov-rp' '(' time ',' [expr]M ',' [expr]N ',' [expr]λON
',' [expr]λOFF ',' [expr]α ',' [expr]Γ ')'
```

XML syntax

```
<extern-function name='markov-rp'>
  time [expr]M  [expr]N  [expr]λON [expr]λOFF [expr]α [expr]Γ
  </extern-function>
```

1.21.3. RDR / Redundancy with Reconfiguration Duration

Description

Albizia offers several functions to calculate the reliability and the availability of a set of elements in redundancy m among n. These functions generate a single-phase Markov graph to perform the calculations (cf. Section 1.20, “MKVM / Markov matrix”).

This configuration is characterized by an interruption of service in the event of failure of an active element during the entire duration of the Treconf reconfiguration.

Parameters	*M* : Number of functional elements required to perform the function, *N* : Total number of items available, *Lambda On (λON)* : Element failure rate when the equipment is turned on, *Lambda Off (λOFF)* : Failure rate of an element when the equipment is switched off, *Alpha (α)* : The use rate α corresponds to the operating time of equipment over the total time of the mission. *Gamma (Γ)* : Probability of failure on demand. *Reconfiguration delay (Treconf)* : Average switching time on one of the redundant elements
Definitions	$\lambda = \lambda_{Active} = \alpha * \lambda_{ON} + (1- \alpha) * \lambda_{OFF}$ $\lambda^* = \lambda_{OFF}$ tr = 1/Treconf At t = 0 the probability of being in the nominal state is 1- Γ At t = 0 the probability of being in the failure state (System KO) is Γ The "System KO" state is a state where the system is unavailable. All "reconfiguration" states are also considered to be states where the system is unavailable. The system is available in other states.

Transition matrix

	0	**1**	**2**	**3**	**4**	**...**		**2(N-M)**	**2(N-M)+1**
No failure : 0	-	Mλ	(N-M)λ*						
Reconfiguration : 1		-	tr	(M-1)λ+(N-M)λ*					
Loss of 1 element : 2			-	Mλ	(N-M-1)λ*				
Reconfiguration : 3				-	tr				
Loss of 2 elements : 4					-				
...									
Reconfiguration : 2(N-M)-1							-	tr	(M-1)λ+λ*
Loss of N-M elements : 2(N-M)								-	Mλ
System KO : 2(N-M) +1									-

Textual syntax

```
'markov-rdr' '(' time ',' [expr]M ',' [expr]N ',' [expr]λON
',' [expr]λOFF ',' [expr]α ',' [expr]Γ ',' [expr]Treconf ')'
```

XML syntax

```
<extern-function name='markov-rdr'>
  time [expr]M  [expr]N  [expr]λON [expr]λOFF [expr]α [expr]Γ
[expr]Treconf
  </extern-function>
```

1.21.4. RER / Redundancy of Repairable Elements

Description	*Albizia* offers several functions to calculate the reliability and the availability of a set of elements in redundancy m among n. These functions generate a single-phase Markov graph to perform the calculations (cf. Section 1.20, “MKVM / Markov matrix”).

This configuration is characterized by the possibility of repairing an element taking into account its MDT. This function considers only one repairer.

Parameters

M Number of functional elements required to perform the function,
N Total number of items available,

Lambda On (λON) Element failure rate when the equipment is turned on,

Lambda Off (λOFF) Failure rate of an element when the equipment is switched off,

Alpha (α) The use rate α corresponds to the operating time of an equipment over the total time of the mission.

Gamma (Γ) Probability of failure on demand,
MDT (Mean Down Time) Mean down time (detection + repair or standard exchange)

Definitions

$\lambda = \lambda_{Active} = \alpha * \lambda_{ON} + (1- \alpha) * \lambda_{OFF}$

$\lambda^* = \lambda_{OFF}$

$\mu = 1/MDT$

At t = 0 the probability of being in the nominal state is 1 - Γ

At t = 0 the probability of being in the failure state (KO) is Γ
The "System KO" state is a state where the system is unavailable. The system is available in other states.

Transition matrix

	0	**1**	**2**	**...**	**N-M**	**N-M+1**
No failure : 0	-	Mλ + (N-M)λ*				
Loss of 1 element : 1	μ	-	Mλ + (N-M-1)λ*			
Loss of 2 elements : 2		μ	-			
...						
Loss of N-M elements : N-M					-	Mλ
System KO : N-M+1					μ	-

Textual syntax

```
'markov-rer' '(' time ',' [expr]M ',' [expr]N ',' [expr]λON
',' [expr]λOFF ',' [expr]α ',' [expr]Γ ',' [expr]MDT ')'
```

XML syntax

```
<extern-function name='markov-rer'>
  time [expr]M  [expr]N  [expr]λON [expr]λOFF [expr]α [expr]Γ
[expr]MDT
  </extern-function>
```

1.21.5. RRR / Redundancy Repairable with Reconfiguration Duration

Description

Albizia offers several functions to calculate the reliability and the availability of a set of elements in redundancy m among n. These functions generate a single-phase Markov graph to perform the calculations (cf. Section 1.20, “MKVM / Markov matrix”).

This configuration is characterized by an interruption of the function throughout the duration of the reconfiguration in the event of failure of an active element, considering that the elements are repairable.

Parameters

M Number of functional elements required to perform the function,
N Total number of items available,

Lambda On (λON) Element failure rate when the equipment is turned on,

Lambda Off (λOFF) Failure rate of an element when the equipment is switched off,

Definitions

Alpha (α) The use rate α corresponds to the operating time of an equipment over the total time of the mission.

Gamma (Γ) Probability of failure on demand,
Reconfiguration delay (Treconf) : Average switching time on one of the redundant elements
MDT (Mean Down Time) Mean down time (detection + repair or standard exchange)

$\lambda = \lambda_{Active} = \alpha * \lambda_{ON} + (1- \alpha) * \lambda_{OFF}$

$\lambda^* = \lambda_{OFF}$

tr = 1/Treconf

$\mu = 1/MDT$

At t = 0 the probability of being in the nominal state is 1 - Γ

At t = 0 the probability of being in the failure state (KO) is Γ

The "System KO" state is a state where the system is unavailable. All "reconfiguration" states are also considered to be states where the system is unavailable. The system is available in other states.

Transition matrix

	0	**1**	**2**	**3**	**4**	**...**		**2(N-M)**	**2(N-M)+1**
No failure : 0	-	Mλ	(N-M)λ*						
Reconfiguration : 1	μ	-	tr	(M-1)λ+(N-M)λ*					
Loss of 1 element : 2	μ		-	Mλ	(N-M-1)λ*				
Reconfiguration : 3			μ	-	tr				
Loss of 2 elements : 4			μ		-				
...									
Reconfiguration : 2(N-M)-1							-	tr	(M-1)λ+λ*
Loss of N-M elements : 2(N-M)								-	Mλ
System KO : 2(N-M) +1								μ	-

Textual syntax

```
'markov-rrr' '(' time ',' [expr]M ',' [expr]N ',' [expr]λON
',' [expr]λOFF ',' [expr]α ',' [expr]Γ ',' [expr]Treconf
',' [expr]MDT ')'
```

XML syntax

```
<extern-function name='markov-rrr'>
  time [expr]M  [expr]N  [expr]λON [expr]λOFF [expr]α [expr]Γ
[expr]Treconf [expr]MDT
  </extern-function>
```

1.22. OCC / Occurrences of failures

The Failure rate is calculated divided the numbers of observed failures by the observation period. The result is a constant law.

Parameters:

- **Number of failures**
- **Period** (Duration) = observation duration

The law is defined as follows:

$$Q(t) = \frac{n}{\tau}$$

1.23. SIL / SIL level

This law corresponds to a constant law with parameter Q = $1x10^{-(SIL-Epsilon)}$

$$Q(t) = 1x10^{-(SIL - \varepsilon)}$$

1.24. RRF / Risk Reduction Factor

This law corresponds to a constant law with a parameter Risk reduction Factor (RRF)

$$Q(t) = \frac{1}{RRF}$$

1.25. EXP / Expression

Law is defined buy user with an Albizia expression that contains time(). An Albizia expression can contain several operators and functions (*, +, -, /, gamma(), exp(), sqrt(), min(), pow(), sin(), ...).
Parameters:

- **Q(t)**: expression to evaluate Probability (must contained time());
- **w(t)**: expression to evaluate unconditional failure rate (must contained time()).

1.26. STO / Stored Electrical Component

This law corresponds to a constant law for stored electrical components according to the functioning time and the storage time on the mission time.

The result of the computation will be a constant probability calculated at the end of the mission.

Parameters :

- **Lambda** (rate) = failure rate
- **Tf** (Duration) = yearly functioning time
- **Ts** (Duration) = yearly storage time
- **K** (Rate) = reduction coefficient (functioning failure rate is equal to storage failure divided by this coefficient)
- **%FMD** (Ratio) = failure mode ratio
- **mission time** (Duration) = duration of the mission

The law is defined as follows :

$$Q(t) = \left(1 - e^{(-\lambda.TTf.\%FMD)}\right) + \left(1 - e^{(-\left(\frac{\lambda}{K}\right).TTs.\%FMD)}\right)$$

Mission time is taken into account in the computation of total functioning time (TTf) and total storage time (TTs) for the electrical component

TTf is defined as follows :

$$TTf = Tf \cdot \frac{DM}{8760}$$

and TTs is defined as follows :

$$TTs = Ts \cdot \frac{DM}{8760}$$

Assuming the number of hours in a year is set to 8760.

2. Uncertainties on the parameters

For each probability law used in the model, it is possible to introduce an uncertainty on each of the parameters. There are several laws available to model them:

- "UNIF / Uniform";
- "NORM / Normal";
- "NLOG / Lognormal";
- "OBS / Observation";
- "OBS (#) / Periodic Observation" ;
- "GAM / Gamma";
- "BET / Beta";
- "TRI / Triangular".
- "HST / Histogram".

Using this method, it is thus possible to introduce the impact of the uncertainties on the data into the final result.

2.1. UNI / Uniform law

This law has two parameters: and upper limit and a lower limit.

Parameters:

- **a** = upper limit
- **b** = lower limit

The law is defined as follows:

$$Q(t) = \frac{(t-a)}{(b-a)}$$

2.2. NLOG / Log normal law

This law has 3 parameters: the mean and the error factor and the percentage of confidence interval.

Parameters:

- **Average(Mu)** = The average
- **Error factor** = The error factor EF (= exponential(1.645*Sigma) for a 90% confidence interval)
- **Confidence interval at** = Percent of confidence interval (between 0 and 1)

A random variable is distributed according to a lognormal distribution if its logarithm is distributed according to a normal distribution. The law is defined as follows:

$$Q(t) = 1 - \int_0^t f(t)dt \qquad f(t) = \frac{1}{t\sigma\sqrt{2\pi}} e^{-\left(\frac{(\ln t - \mu)^2}{2\sigma^2}\right)}$$

Where Sigma is equal to ln(EF)/coef, where coef is the quantile of the normal law conresponding to the chosen percentage (1.645 for 90%), and where Mu = ln(E(x)) - SigmaÂ²/2

2.3. NORM / Normale

This law has two parameters: the mean and the standard deviation.

Parameters:

- **Mu** = mean
- **Sigma** = standard deviation

The law is defined as follows:

$$Q(t) = 1 - \int_0^t f(t)dt \qquad f(t) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\left(\frac{(t-\mu)^2}{2\sigma^2}\right)}$$

2.4. OBS / Observation

This law has two parameters.

Parameters:

- **Number of events (N)** = Number of events observed
- **Observation duration (T)** = Observation duration

The probability density function of this distribution is:

$$f_x(T) = \frac{1}{2^{\frac{k}{2}}\,\Gamma(\frac{k}{2})}\, t^{\frac{k}{2}-1} e^{-\frac{T}{2}}$$

$$\text{With } \Gamma(z) = \int_0^{+\infty} t^{z-1} e^{-t} dt \;\; the\ Gamma\ function$$

k represents the degrees of freedom.

In options, it is possible to choose the degrees of freedom

2.5. OBS (#) / Periodique Observation

This law has three parameters, it is based on F.Brissaud work published in Rel. Eng. Sys. Safety 2017 DOI:10.1016/j.ress.2016.11.003

Parameters:

- **Number failure revealed (N)** = Total number of failure observed
- **Duration between 2 tests (#)** = Inspection period

- **Number of proof tests (W)** = Total number od proof tests.

This function is partly based on a random number generator that uses a beta distribution (W-N + 1, N).

2.6. GAM / Gamma

The gamma distribution is a two-parameter probability distributions: the shape parameter and the scale parameter.

Parameters:

- **K** = Shape parameter
- **Theta (θ)** = Scale parameter

The probability density of the gamma distribution is:

$$f(x) = x^{k-1} \frac{e^{-x/\theta}}{\theta^k \Gamma(k)}$$

$$\text{With } \Gamma(z) = \int_0^{+\infty} t^{z-1} e^{-t} dt \ \ the\ Gamma\ function$$

2.7. BET / Beta

The beta distribution is parametrized by two positive shape parameters: Alpha et Beta.

Parameters:

- **Alpha (α)** = Shape parameter
- **Beta (β)** = Shape parameter

The probability density function for $0 \leq x \leq 1$, and shape parameters $\alpha, \beta > 0$ is :

$$f(x; \alpha, \beta) = \frac{1}{B(\alpha, \beta)} x^{\alpha-1} (1 - x)^{\beta-1}$$

$$B(x, y) = \int_0^1 t^{x-1} (1 - t^{y-1}) dt$$

2.8. TRI / Triangulaire

This law has three parameters : a minimum, a maximum et an optimum.

Paramètres :

- **a** = minimum
- **b** = maximum
- **c** =optimum

The law definition is:

$F_c = \frac{c-a}{b-a}$:used during Z testing

$p_1 = \sqrt{(c-a)(b-a)}$:gradient between a and c

$p_2=\sqrt{(b-c)(b-a)}$: gradient between c and b

In propagation uncertainties:

Z randomly fired and equidistributed distributed between 0 and 1;

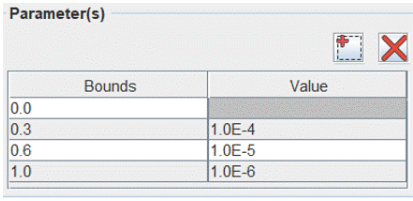
si $Z=0$
  $d=a$
sinon
  si $Z<F_c$
    $d=a+p_1\sqrt{Z}$
  sinon
    si $Z<1$
      $d=b-p_2\sqrt{1-Z}$
    sinon $d=b$

2.9. HST / Histogramme

Draw a random number between the minimal bound and the maximal bound, and return value corresponding to the interval containing the value. the law has as many parameters as desired bound.

The law definition is :

- **Bounds** = bound of the value in the histogram.
- **values** = Value between two bounds. the two corresponding bounds are [A;B], where A is the bound located in the row before the current value and B the bound located on the same row of the value. The value on the first row is always empty, since the firts bound is used as the minimal bound of the value on the second row.

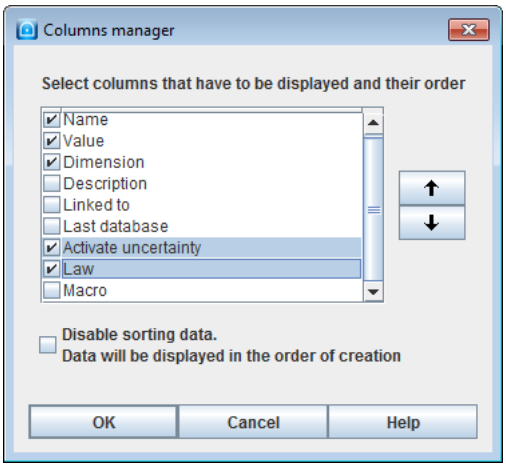


2.10. Consideration of the uncertainties

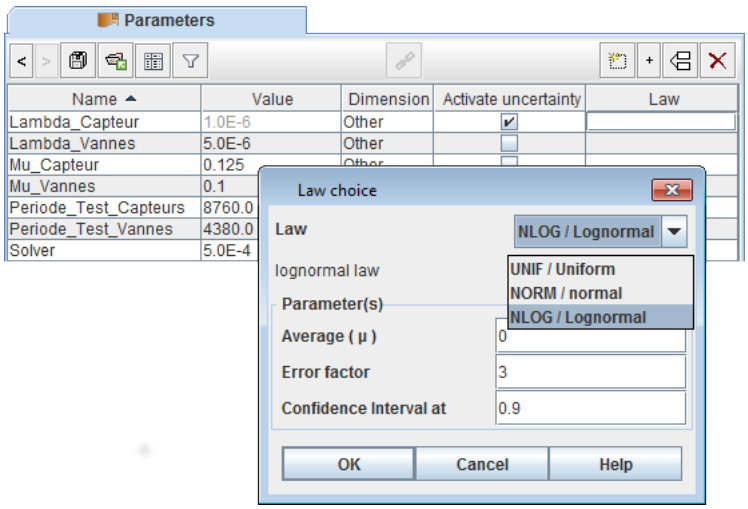
Uncertainties on parameters can be evaluated with 2 different ways:

- in the configuration of the laws as described in the chapter Configuring the laws
- in the tab of parameters. In this case, ff a same parameter is used in 2 events, the same uncertainty is considered

  To do that, first it is necessary to select the column **Activate uncertainty** and **Law** using the columns manager.



After, in the tab of parameters, It is enough to choose or not to activate the uncertainties and the law will be applied in **Law** column.



F. Glossary

1. Format

All values can be entered in two different ways:

- **"Normal" notation**: the decimal separator is the dot, e.g. 0.0000015.
- **Scientific notation**: the decimal separator is the dot, e.g. 1.5E-6 which corresponds to 0.0000015.

2. Definition and explanation of the acronyms and parameters

SFF (Safe Failure Fraction): corresponds to the safe failure rate ($\lambda_{sd} + \lambda_{su} + \lambda_{dd}$) / λ

– **Vote with "A" type architecture**: the invalidity of the sensor triggers no action other than an alarm (availability). The solver logic is modified, excluding sensors with detected failure. In this case, we define a number (X) of detected failure from which the channel trips. This number (X) is fixed by default for TotalEnergies (but can be modified in M configuration):
  3 if 3 components or more
  2 if 2 components
  1 if 1 component

– **Vote with "S" type architecture**: the invalidity of the sensor triggers the safety system (Safe).

- **Beta (β)**: proportion of common cause failures (in %).
- **CCF or DCC**: Common Cause Failure. When several identical elements are put in a system, there is always a probability that they will fail at the same time from a common cause (design problem, external phenomena for example). This is called a common cause failure.
- **Component available during test (X)**: specifies whether the component is able to carry out its safety mission during the test (if the checkbox is checked).
- **DC**: on-line diagnostic coverage and is a rate between 0 and 100%. A 0% rate means that no revealed failure can be detected.
- **DCd**: on-line diagnostic coverage of dangerous failures and is a rate between 0 and 100%. A 0% rate means that no revealed dangerous failures can be detected.
- **DCs**: on-line diagnostic coverage of safe failures and is a rate between 0 and 100%. A 0% rate means that no revealed safe failures can be detected.
- **Detected**: applies to the equipment and means detected by diagnostic tests, periodic tests or human intervention (e.g. physical inspection and manual tests) or during normal operation.
- **DC only alarmed** :Percentage of detected failure that are only alarmed (non-triggering). This field is available only if channel is in M Mode.
- **DCS**: Distributed Control System
- **Determinate**: A component can be one of these 3 types: "Non-type A/B", "Type A" or "Type B"
- **Duration between tests (T1)**: period of time between two proof tests of the component.
- **E/E/PE**: electrical/electronic/ programmable electronic. Technology based on electricity (E), and/or electronics (E) and/or programmable electronics (EP). NB. - This term designates all devices which work according to electrical principles.
- **Proportion of detected failure** :proportion of hidden failures detected during partial stroking tests (0-100%). 0% means no failure is detected, 100% means every failure is detected.
- **Failure**: a functional unit ceases to accomplish its required function.
- **Lambda λ**: failure rate of the component (h^{-1}).
- **Lambda D (λ d)**: dangerous failures. Failure with the potential to put the safety system into a dangerous state or make it unable to carry out its function.
- **Lambda DU (λ_{du})**: Dangerous undetected failure rate of the component (h^{-1}).
- **Lambda DD (λ_{dd})**: Dangerous detected failure rate of the component (h^{-1}).
- **Lambda S (λ s)**: safe failures. Failure with the potential to put the safety system into a safe state in carry out its function.

- **Lambda SU (λ_{su})**: Safe undetected failure rate of the component (h^{-1}).
- **Lambda SD (λ_{sd})**: Safe detected failure rate of the component (h^{-1}).
- **Lambda during test λ***: failure rate of the component during the test (h^{-1}). The test conditions may cause extra stress and increase the lambda.
- **MDT (in h)**: indicates the mean time between the occurrence of a failure and the re-start of the system (Mean Down Time). It is the average downtime.
- **MTTF (in h)**: indicates the mean time between the start-up of the system and the occurrence of the first failure (Mean Time To Failure). It is the average time of operation before the first failure occurs. λ = 1/ MTTF for a component.
- **MTTR (Mean Time To Repair) in h**: mean time between detection of a failure and the repair of the component.
- **Non-detected (Undetected)**: applies to the equipment and means non-detected by diagnostic tests, periodic tests or human intervention (e.g. physical inspection and manual tests) or during normal operation.
- **Number of tests**: number of partial stroking tests carried out between two full tests.
- **Operating duration (Years)**: means the foreseen operating industrial duration of the Safety Instrumented Function (SIF) installed on its process unit.
- **PFD** : Probability of Failure on Demand. Cf. Norm IEC61508. Can be defined as Unavailability
- **PFH** : Probability of Failure per Hour. Cf. Norm IEC61508. Can be defined as Unconditionnal Failure Intensity
- **Redundancy** : implementation in parallel of elements which have the same safety function so that the sub-system is more available.
- **Repair rate μ (Mu)**: repair rate in $^{-1}$, whose symbol is (μ). This value is equal to 1/MTTR, for a repair time of 48h, Mu = 1/48 = 2.08E-2
- **Switch time parameter**: is the period of time during which the component causing the failure is disconnected from the system and replaced by a component in working order. This time is necessarily lower than the MMTR.
- **R.R.F** : Risk Reduction Factor of the SIF
- **Safety function**: function to be carried out by an E/E/EP safety system, by a safety system based on another technology or by an external risk reduction device, designed to ensure or maintain the controlled system in a safe state with regard to a specific dangerous event.
- **SIF** : Safety Instrumented Function.
- **SIL 0**: instantaneous PFD $\in$ [10^{-1}; 1]. instantaneous PFH $\in$ [10^{-5}; +infinity].
- **SIL 1**: instantaneous PFD $\in$ [10^{-2}; 10^{-1}[. PFH instantanée $\in$ [10^{-6}; 10^{-5}[.
- **SIL 2**: instantaneous PFD $\in$ [10^{-3}; 10^{-2}[. instantaneous PFH $\in$ [10^{-7}; 10^{-6}[.
- **SIL 3**: instantaneous PFD $\in$ [10^{-4}; 10^{-3}[. instantaneous PFH $\in$ [10^{-8}; 10^{-7}[.
- **SIL 4**: instantaneous PFD $\in$ [0; 10^{-4}[. instantaneous PFH $\in$ [0; 10^{-8}[.
- **SIS**: Safety Instrumented System. Instrumented system used to carry out one or several safety functions. An SIS is made up of sensors, a logical processing system and actuators.
- **System**: set of elements which interact according to a specific model, an element, which may be another system called a sub-system. The sub-systems can themselves be either a command system or a controlled system made up of hardware, software and interacting with man.
- **S-PLC**: Safety-Programmable Logic Controller
- **Test duration π (Pi)**: period of time necessary for testing the component.
- **Test efficiency rate σ (Sigma)** : cover or efficiency rate of the test. The value ranges from 0 (the test never detects anything) to 1(the test always detects the failure).
- **Test leads to failureγ (Gamma)**: probability [0,1] that the test will cause the hardware to fail. 0 means no test causes any failure, 1 mean every test causes failures.
- **Test when unit is stopped**: means that the component is tested when the unit is stopped. The test does not harm the safety function as the unit is no longer working.
- **Test when unit is working**: means that the component is tested when the unit is working. The component is no longer available to carry out its function and this affects the safety function. This can be used when a sensor has been by-passed to be tested and the installation has not been stopped.
- **Time of the first test (T0)**: time at which the first test of the component is carried out.
- **Wrong re-setup after testsω1 (Omega1)**: probability [0,1] of wrong re-setup of the equipment after the test. It is the probability that the component will not be able to carry out its safety mission after being tested by the operator. It can be left at 0 if you consider that the operators and test procedures are infallible (no omission of a by-passed sensor, powering up the motor, etc.).

- **Wrong re-setup after repairsω2 (Omega2)**: probability [0,1] of wrong re-setup of the equipment after the repairs. It is the probability that the component will not be able to carry out its safety mission after being repaired (or changed) by the operator. It can be left at 0 if you consider that the operators and repairs procedures are infallible (powering up the new motor, etc.).