

# Step by step simulator

## *System simulation*

---

Copyright © 2007 Dassault Systemes

Version 1.0

### *Abstract*

Simulation module enables to simulate system behavior in an interactive way, depending on events that are triggered on components.

Simulation enables validation of behavior of components or equipments that are in system architecture.

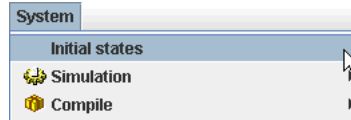
# Table of Contents

<b>Simulation configuration .....</b>	<b>4</b>
Set initial value of state variables up .....	4
Events configuration .....	5
<b>Simulation launch .....</b>	<b>7</b>
<b>Simulating .....</b>	<b>9</b>
Commands available in simulation .....	9
Move along equipments/systems view .....	10
<b>Transitions triggering .....</b>	<b>11</b>
<b>State variable modification .....</b>	<b>12</b>
<b>Simulator display window .....</b>	<b>13</b>
Generalities .....	13
Transition list .....	13
Variable list .....	14
Hierarchical view - Tree .....	15
Select view .....	15
Transition History .....	16
Scheduler .....	16
Sequences .....	17
<b>Other fonctionnalities during simulation .....</b>	<b>20</b>
Initial configuration saving during simulation .....	20
Search for component during simulation .....	20
To take determinist events into account .....	22
Introduction .....	22
Consideration in scheduler .....	22
Instantaneous transitions .....	22
Configuration of step by step simulator .....	23
Temporal conflict between 2 transitions .....	23

# Simulation configuration

Before simulation launching, initial states have to be defined for simulation.

- Use **System - Initial states** menu in order to define initial conditions.



- **Initialisation** waiting window is displayed.



- Then, initial states window is displayed.

This window enables to chose a new initial state of the system.

Default value of initial state is the set of default value from all component models used in the system.

There are two tabs, one for state variables, the other for events.

## Set initial value of state variables up

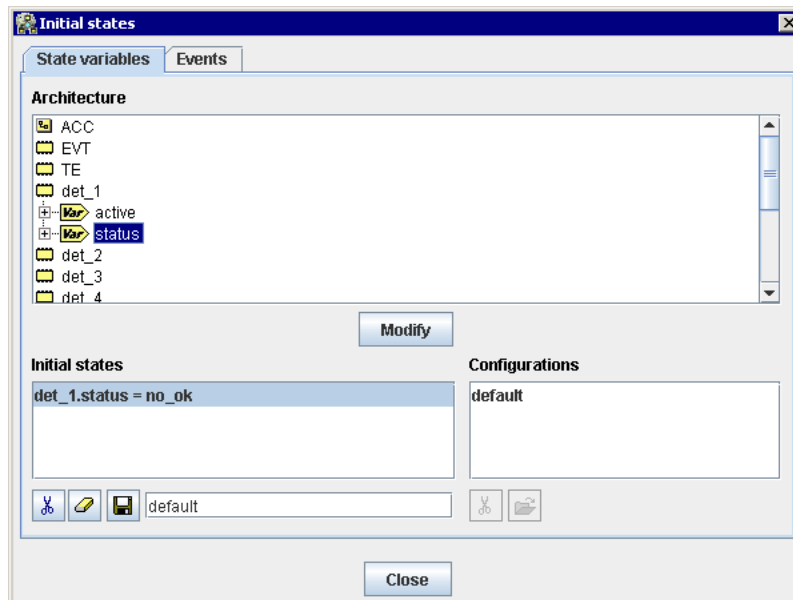


Figure 1. Initial states window - State variables

Default initial state is displayed in red. In order to modify initial state of a variable, double-click on this variable or click on **Modify** button. It will display state variable modification window.

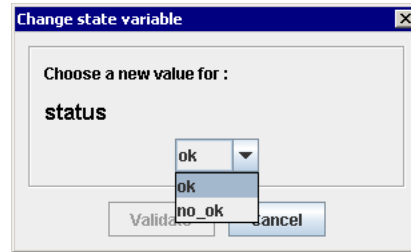


Figure 2. Change state variable

New values are displayed in the initial states frame :

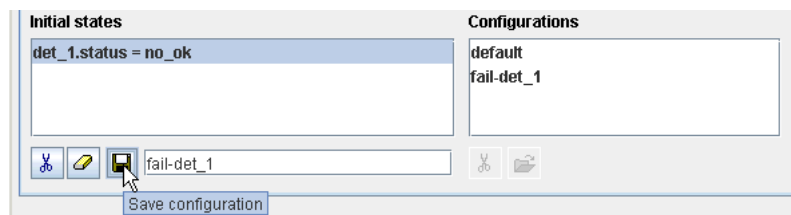







Figure 3. Initial states window - Initial states

- In the field next to , type a name for new configuration (name of the new initial state of the system) then click on  ; the new configuration is saved and is displayed in **Configuration** frame.
- On the left,  icon deletes selected initial value of a variable.  icon deletes all initial values of state variables.
- Click on **Close** button.

Select a previously saved configuration as follows :

- Select a configuration (name of system initial state previously saved).
- Click on  icon to load configuration. Values of state variables are displayed in the left frame : **Initial states**.
- Click on **Close** button.

## Note

These configurations could be used for simulation, and don't depend on type of simulation. The choice of configuration will be made at simulation launch.

## Note

A new configuration can be defined automatically during simulation , Cf. the section called “Initial configuration saving during simulation”.

## Events configuration

Events tab enables to modify events.

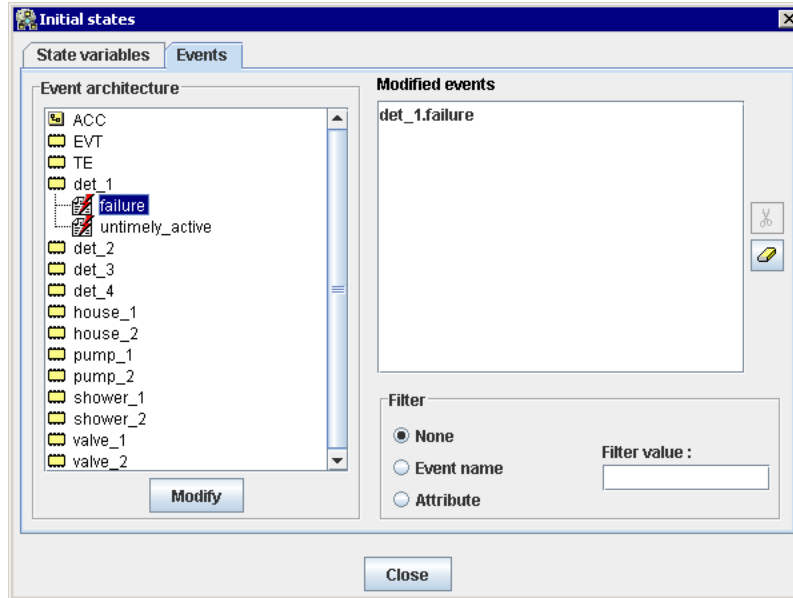


Figure 4. Initial states window - Events

Double click on event or use **Modify** button, it displays event properties window.

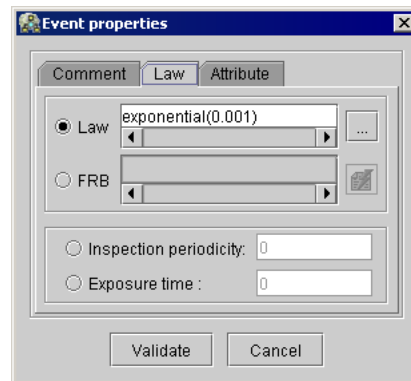



Figure 5. Initial states window - Events

Modified events are displayed on the right. ✂ icon delete selected modified event. 🗑 icon delete all modified events.

# Simulation launch

In order to launch interactive simulation, you can :

- Either use **System - Simulation - Start**
- or click on  icon.

## Note

When many step by step simulator are available, you can choose the one you want by clicking on the little arrow next to the start icon. The selection of a simulator will launch the simulator, and memorise this choice for the next simulation.

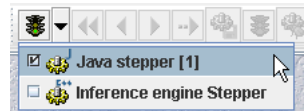


Figure 6. Simulation start icon

Before simulation starts, initial configuration can be chosen.

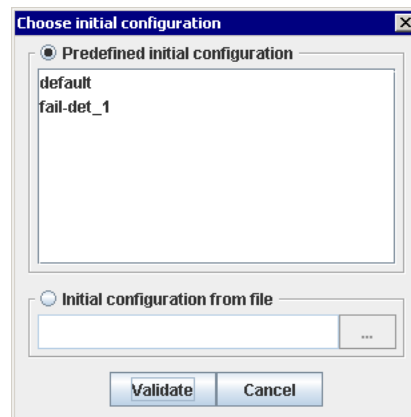


Figure 7. Choose initial configuration

Initial configuration can be :

- Either chosen among previously saved configurations (Cf. the section called “Set initial value of state variables up”).
- or defined externally with a configuration file.

The file format must be a list of couples {name of state variable, initial value of variable} as follows :

```
equipA.compA.Tension = Null
equipA.compB.Tension = Failed
...
```

It corresponds with the following syntax:

```
<init-list>
::= <init-def> ( ' ' <init-def> )*
```

```

<init-def>
  ::= <hierarchy-path> '==' <expression>
  ::= <hierarchy-path> '=' <expression>
<expression>
  ::= true
  ::= false
  ::= <integer>
  ::= <float>
  ::= <identifier>
<identifier> ::= '[a-zA-Z][a-zA-Z0-9_-]*';
<hierarchy-path> ::= <identifier> ('.' <hierarchy-path>)*

```

Each variable in file must be defined in the simulated model.

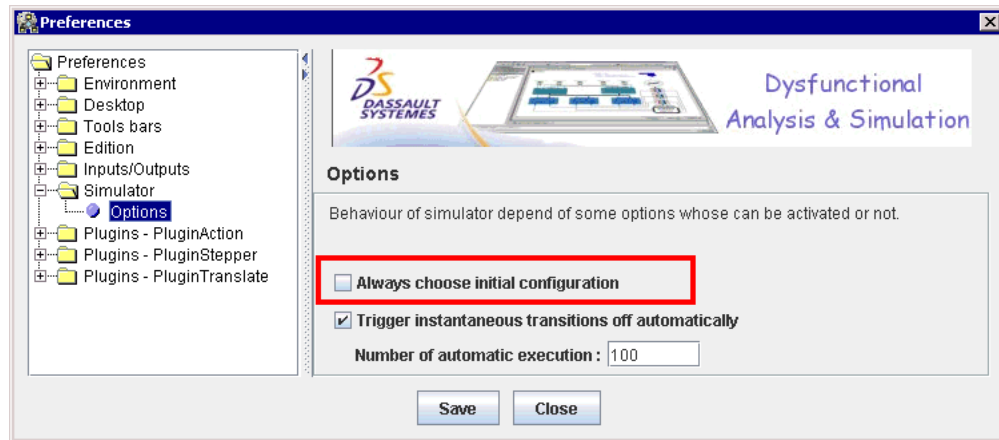
Otherwise, an error message is displayed and current action is stopped.

For initial configuration file, it is also possible to use an xml result file coming from generic sequence generation or FMEA Generation. In this case, initial configuration will be the one defined for generation.

Once configuration is selected, click on **Validate** button to launch simulation.

## Note

The window allowing configuration choice is usually displayed only if there are many previously saved configurations. the display of this window can be forced in software preferences. (**Options - Preferences...** command).



With large systems, launch can take a long time. A waiting window reminds it.



When simulation starts, many verifications are made on every component model in architecture. If an error is detected, a window is displayed and simulation doesn't start.



# Simulating

When simulation starts, initial state of system is displayed: colors of links show value of their variables, icons represent the components in their initial states.

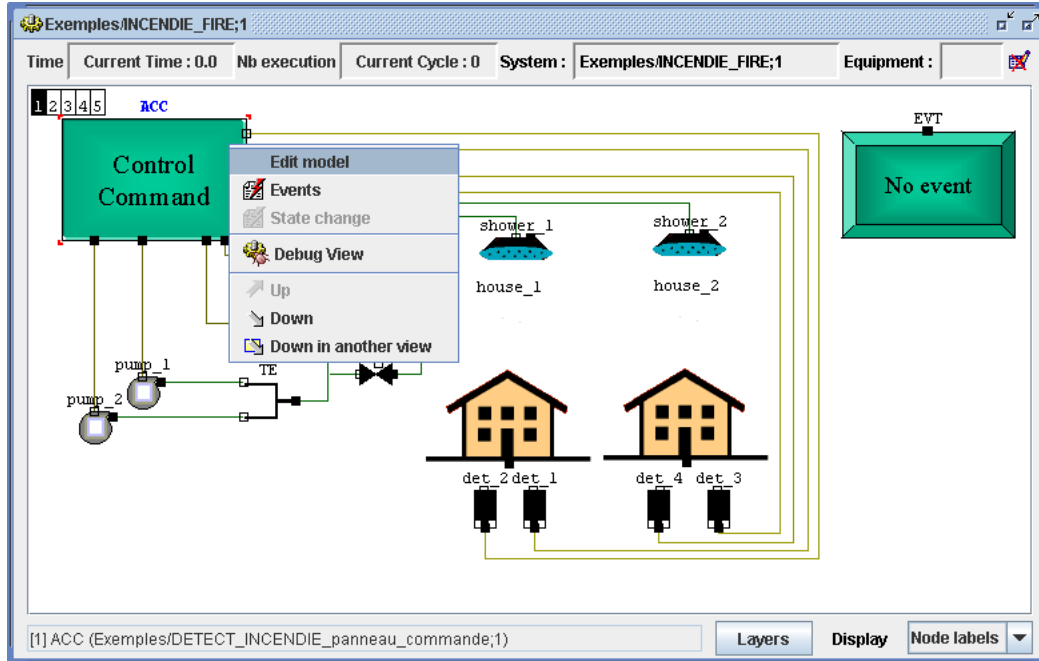
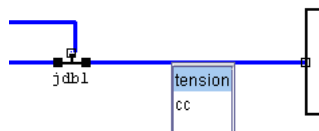


Figure 8. Graphical view during simulation

## Note





when the graphical link represents a bus on which several flow parameters transit, a right click on this link gives access to the flow parameters list in order to modify dynamically the graphical display order of the flow parameters. The link colour resulting from this change of order corresponding to the value of the flow variable located in 1st position



## Commands available in simulation

Following commands are available in simulation:

- **◀◀ (Initialize)**: Reinitialize system in its initial state. This action cancels all event triggering made during simulation,
- **◀ (Go back)**: enables to go back (one step), this action cancel the last triggering of simulation.
- **▶ (Go forward)**: enables to redo (forward) a step that has been previously canceled,
- **→ (Next)**: enables to trigger the next event of scheduler, Cf. the section called “Scheduler”,
- **💾 (Save as initial state)**: enables to memorize current system state during simulation in order to create initial states. Cf. the section called “Initial configuration saving during simulation”,

-  (**Stop**): Stop simulation and close window,
-  (**Informations window**): enables to display information about current simulator (history, scheduler, list of variables, ...); Cf. the section called “Simulator display window”,
-  (**Event selection**): enables to trigger an event/transition of the simulator; Cf. the section called “Transitions triggering”,
-  (**State variable value**): enables to modify state variables of selected element; Cf. the section called “State variable modification”.

### Note

Errors like division by 0 or consistency error that have not been verified on a component can appear during simulation. If such an issue appears, a message is displayed and simulation is stopped.




### Note

Some error that are not definitive can appear, such as a too large number of instantaneous transition triggering.

## Move along equipments/systems view

You can move along graphical view in order to display the system and its sub-equipment.

In order to move along, three icons are available in popup menu and in tool bar:

-  (**Up**): Go to upper hierarchical level
-  (**Down**): Go to lower hierarchical level
-  (**Down in another view**): Display selected event in another view

### Note

A double click on an equipment enables to go **Down**. On components, a double click launch component edition in read-only mode.

# Transitions triggering

## Event triggering /transition firing

Simulation enables to validate a system. System is simulated from its initial state. Then events can appears during life of systems. In AltaRica, action of theses events is defined in transition definitions.

To trigger an event means to trigger (to fire) a transition labeled with this event.

Events/transitions are associated with components. To trigger/fire a transition, select component, then use **System - Simulation - Events** command (or one of its equivalents in toolbar or menu).

If at least one of the transitions associated with component are fireable, the following window is displayed.

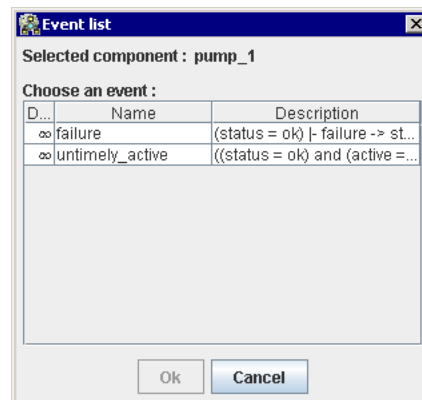


Figure 9. List of component events

The first column displays delay associated with the transition. In case of stochastic (not determinist) events, delay doesn't mean anything. In fact, transition is triggered randomly. The displayed delay is infinite. For determinist transitions see the section called "To take determinist events into account".

The name of the event labeling transition is displayed in the second column, the third one reminds transition definition.

Sort is available on columns (double click on header).

Select a transition and click on **Ok** button to trigger transition.

### Note

In order to trigger a synchronization at equipment level, select the equipment and do the same things. Then, the window displays fireable transitions of equipment and of its sub-components.

### Note

In order to trigger a global synchronization (defined at system level), don't select anything. The window displays all system transitions that are fireable.

# State variable modification

Component state variables can be directly modified. It's mainly used to configure system without transition triggering. If state variables are used as parameters (e.g. for component capacity), it enables to make local modifications in order to see how the system reacts.

Select **Systems - Simulation - State variables** (or one of its equivalents in toolbar or menu) to modify a state variable. The following window is displayed:

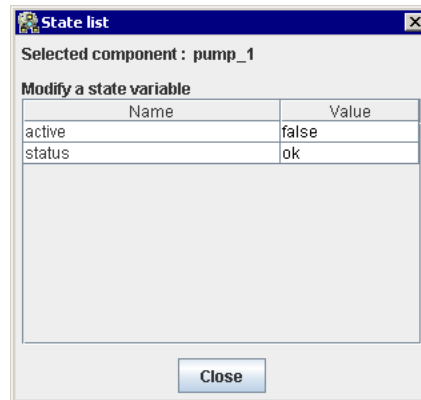


Figure 10. List of component state variables

Double-click on the value to be modified (column **Value** for the considered state variable). The following window displays different available values for the variable.

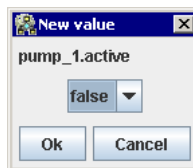


Figure 11. Modification of state variable value

Click on **Ok** button to validate changes.

The simulation result is displayed on screen : component icon changes and colors of some links are modified because of state variable modification.

# Simulator display window

## Generalities

Simulator display window provides further informations than graphical representation.

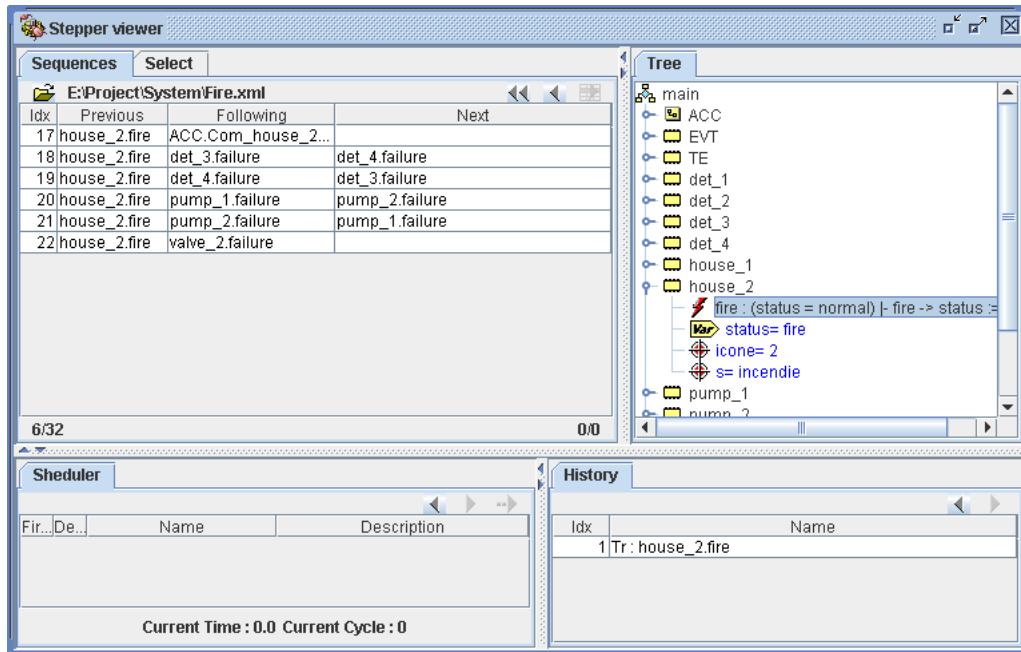


Figure 12. Simulator display

This window is splitted in 4 panels. Each panel can display different views thanks to tabs. View can be moved from tab to tab with a "drag and drop".

Click right on an empty panel or on the header in order to display Contextual menu. It enables to add different types of views in current panel.

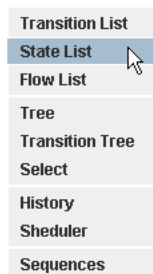
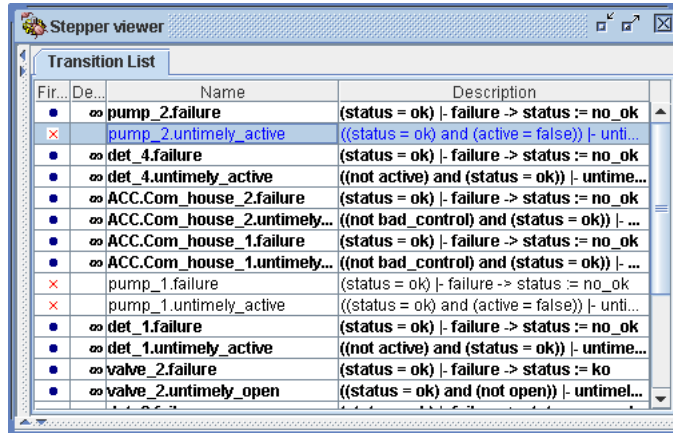


Figure 13. Contextual menu - view creation

Each view has its own functions. Views are described in following chapters.

## Transition list

**Transition list** enables to display fireable or not fireable transitions of current simulation.



Fir...	De...	Name	Description
●	∞	pump_2.failure	((status = ok)  - failure -> status := no_ok
×		pump_2.untimely_active	((status = ok) and (active = false))  - unti...
●	∞	det_4.failure	((status = ok)  - failure -> status := no_ok
●	∞	det_4.untimely_active	((not active) and (status = ok))  - untime...
●	∞	ACC.Com_house_2.failure	((status = ok)  - failure -> status := no_ok
●	∞	ACC.Com_house_2.untimely...	((not bad_control) and (status = ok))  - ...
●	∞	ACC.Com_house_1.failure	((status = ok)  - failure -> status := no_ok
●	∞	ACC.Com_house_1.untimely...	((not bad_control) and (status = ok))  - ...
×		pump_1.failure	((status = ok)  - failure -> status := no_ok
×		pump_1.untimely_active	((status = ok) and (active = false))  - unti...
●	∞	det_1.failure	((status = ok)  - failure -> status := no_ok
●	∞	det_1.untimely_active	((not active) and (status = ok))  - untime...
●	∞	valve_2.failure	((status = ok)  - failure -> status := ko
●	∞	valve_2.untimely_open	((status = ok) and (not open))  - untimel...

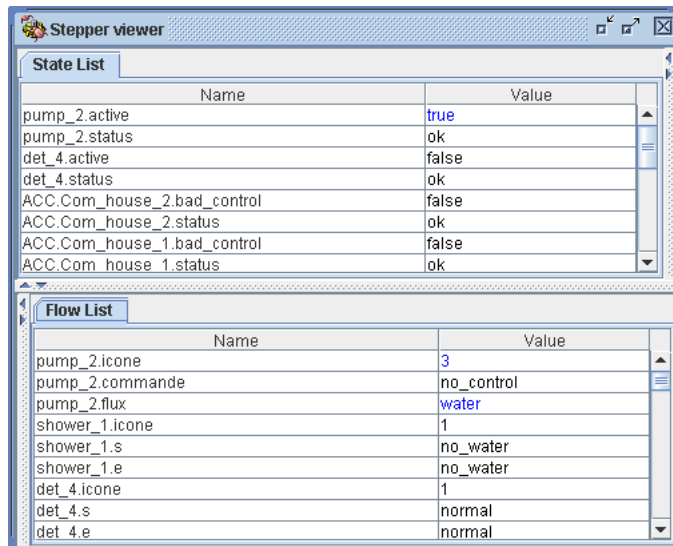
Figure 14. View - Transition list

Columns show :

1. If transition is valid (blue ball) or not valid (red cross).
2. Delay associated with transition. In case of stochastic transitions, delay is infinite.
3. Name of the element associated with transition.
4. Description of transition given by simulator. In case of Java1 Simulator, it is definition of transition.

## Variable list

**State list** (respectively **Flow list**) enables to display state variables (respectively flow variables) of currently simulated model.



State List	
Name	Value
pump_2.active	true
pump_2.status	ok
det_4.active	false
det_4.status	ok
ACC.Com_house_2.bad_control	false
ACC.Com_house_2.status	ok
ACC.Com_house_1.bad_control	false
ACC.Com_house_1.status	ok

Flow List	
Name	Value
pump_2.icone	3
pump_2.commande	no_control
pump_2.flux	water
shower_1.icone	1
shower_1.s	no_water
shower_1.e	no_water
det_4.icone	1
det_4.s	normal
det_4.e	normal

Figure 15. View - State list/Flow list

Table columns can be sorted.

Variables whose value has changed during last firing are displayed in bleu.

Value of a state variable can be directly modified (double-click on the value to be modified).

## Hierarchical view - Tree

**Tree** view enables to see all information during simulation with a hierarchical way.

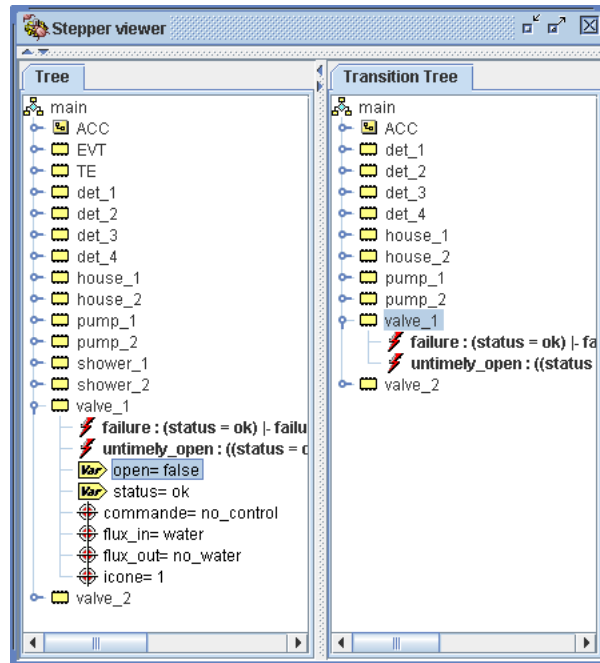


Figure 16. View - [Transitions] Tree

**Transitions Tree** displays all transitions with a hierarchical way. If a component/equipment doesn't have any transitions, it is not displayed in the list.

Transitions are in bold, variables whose value has changed during last step are displayed in blue.

Selection of component/equipment or change of hierarchical level in graphical simulator leads to selection of the node in the tree.

## Select view

**Select** displays tree related to the selected graphical component.

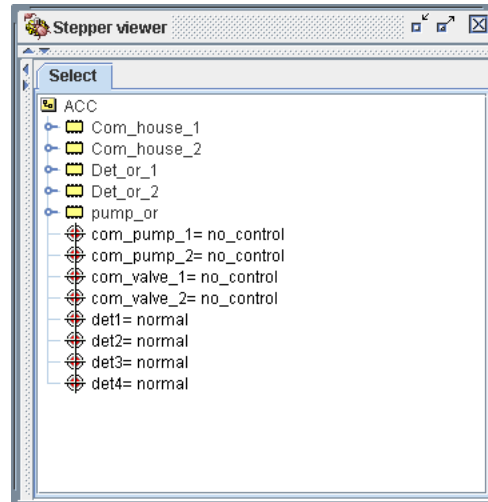


Figure 17. View - Select

## Transition History

**History** enables to see which transition have already been fired/triggered.

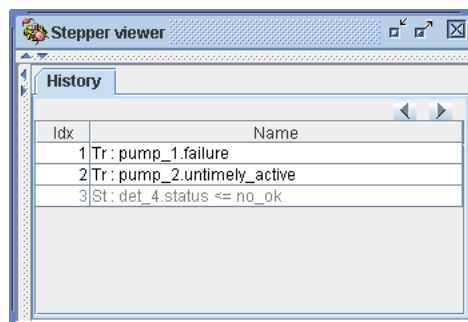


Figure 18. View - History

When the end of the list is in gray, it means that user has gone back with **Go Back** [<] command ◀. History can be redone with **Go forward** [>] command ▶.

A double-click on a transition enables to go forward to the state before selected transition triggering.

If a state variable is modified, it is displayed in history.

## Scheduler

**Scheduler** displays determinist transitions that are valid. They are sorted following the order in which they will be fired in step by step simulator.



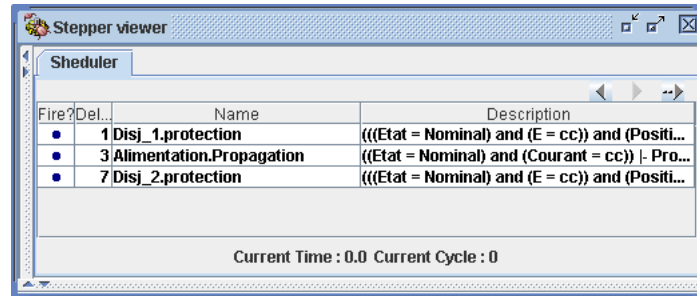


Figure 19. View - Scheduler

→ **Next** command triggers the first transition of scheduler.

For more informations about determinist transitions, Cf. the section called “To take determinist events into account”

## Sequences

**Sequences** enables to graphically replay sequences coming from sequence generation or from other tools.

You can :

- Validate sequences (functions of search algorithm, some sequences may not lead to wanted result)
- Help user to understand meaning of sequences (in a model behavior point of view), with graphical display (link colors, icons) and textual display (state/flow variable value).

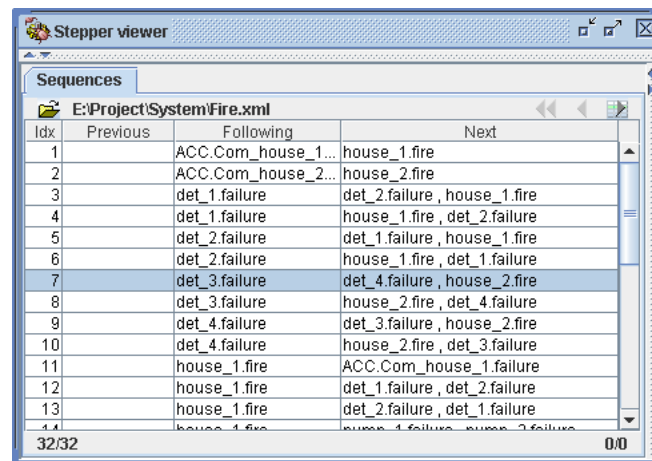


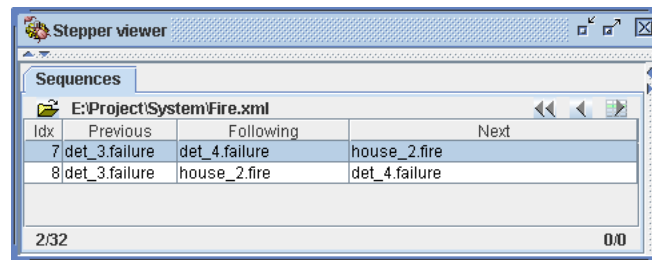
Figure 20. View - Sequences

Each sequence is represented by a table line.

Columns give:

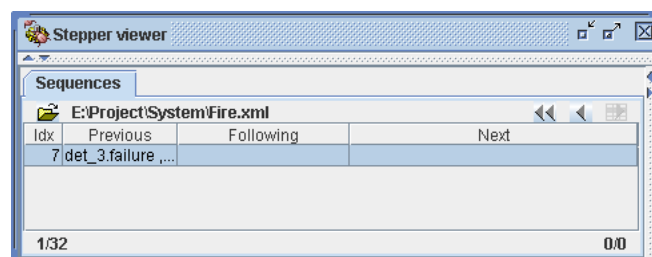
1. ID or number of the sequence.
2. **Previous** event of the sequence, that is to say events already fired.
3. **Following** fireable event, that is to say the following event that will be fired for the selected sequence.
4. Remaining event (the end of sequence).

Select  button to skim through the sequence. The **following** event is fired. A double-click on sequence or pressing space bar have the same effect.



List is updated functions of current sequence (i.e event history).

Repeat to skim sequence through the end.



**Sequences** is linked to a set a sequences coming from a file. To load a sequence file, use  button.

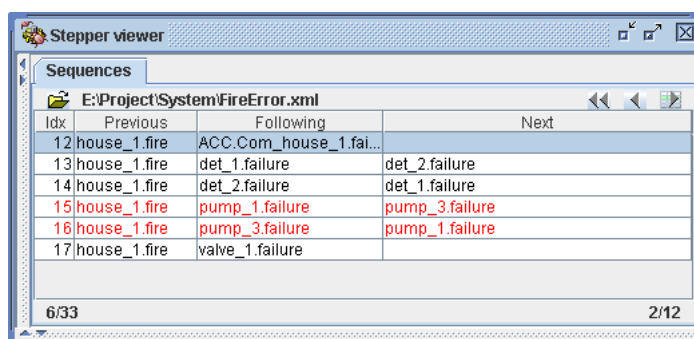
File must be in XML format of generic sequence generator (Cf appendix on generic tools for more informations).

It must fit to the following format:

```
<?xml version='1.0'?>
<seqgen>
...
<result>
  <seq><tr evt="synthesys_2.failure"/>
    <tr evt="screen_1.failure"/>
    <tr evt="screen_pilot.and_2.out_1"/></seq>
  <seq><tr evt="synthesys_2.failure"/>
    <tr evt="screen_1.failure"/>
    <tr evt="screen_pilot.relay_2.untimely_close"/></seq>
...
</result>
...
</seqgen>
```

Presence of events in the model is verified during file loading.

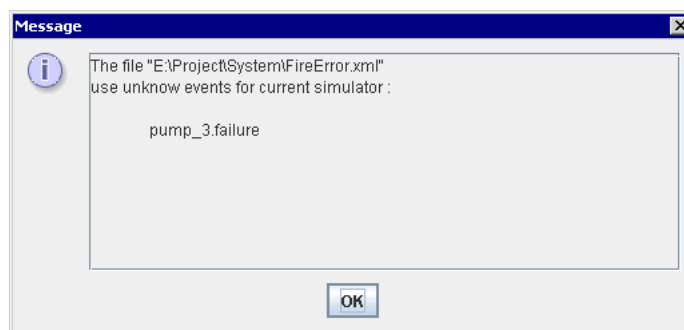
If an error happens,



Idx	Previous	Following	Next
12	house_1.fire	ACC.Com_house_1.fai...	
13	house_1.fire	det_1.failure	det_2.failure
14	house_1.fire	det_2.failure	det_1.failure
15	house_1.fire	pump_1.failure	pump_3.failure
16	house_1.fire	pump_3.failure	pump_1.failure
17	house_1.fire	valve_1.failure	

6/33 2/12

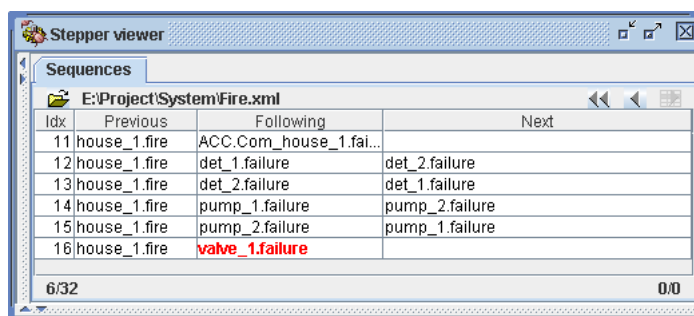
- Sequences having issues are displayed in red
- Number of these sequences is displayed in the bottom right.
- To display errors, double-click on the bottom right.



## Note

For each simulation step, sequences of the current list are validated: one and only one transition labelled with following event must be fireable.

Otherwise, no valid transition or more than one, next event is displayed in red (bold or italic) and sequence can not be played.



Idx	Previous	Following	Next
11	house_1.fire	ACC.Com_house_1.fai...	
12	house_1.fire	det_1.failure	det_2.failure
13	house_1.fire	det_2.failure	det_1.failure
14	house_1.fire	pump_1.failure	pump_2.failure
15	house_1.fire	pump_2.failure	pump_1.failure
16	house_1.fire	<b>valve_1.failure</b>	

6/32 0/0

In the left bottom corner, current number of sequences and total number are displayed.

# Other fonctionnalities during simulation

## Initial configuration saving during simulation

During simulation, current state can be saved as initial configuration.

This command is available in **System - Simulation - Save as initial state** menu, or with  icon.

When command is selected, following window is displayed:

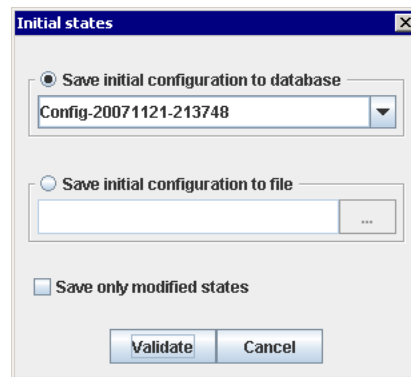


Figure 21. Initial state saving


You can save current state as initial configuration:

- Either as pre-saved configuration of the system: In this case, validate or modify the proposed name. If you choose an existing configuration name, it will be erased. You can see saved initial configuration in **System - Initial states** menu. (Cf. the section called “Set initial value of state variables up”).
- Or in a text file whose format is defined in simulation launching section [7].

The **Save only modified states** checkbox enables to keep only differences with default initial state of the system.

## Search for component during simulation

During simulation on a system architecture, the search for an instance of component is made as follows:

- Use **Edit - Rechercher** command (Ctrl+B) or  icon in edit toolbar. Then, **Search** window is displayed.
- In a system, click on **Component** tab. In **Filter** field, type the name (case sensitive) of searched component as follows: [Equipement\_Name].[Component\_Name]

The '\*' character can be used to limit search, or to complete name of a hierachical level or a component name (examples : HYD\*.Y\_VF\*, \*.Y\_VF, \*Y\_VF\*).

- Click on  icon. The searched component (or components having name matching filter) is (are) displayed in the lower part of the window.

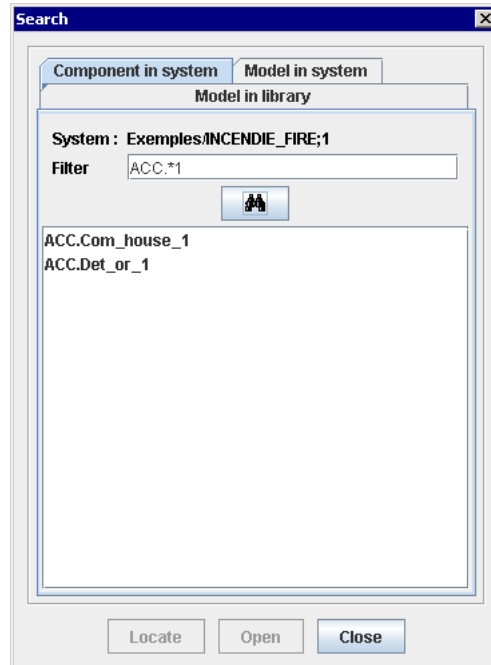


Figure 22. Search component in a system

- Select searched component.
- Click on **Locate** button. The hierarchical level containing component is displayed and component is selected.

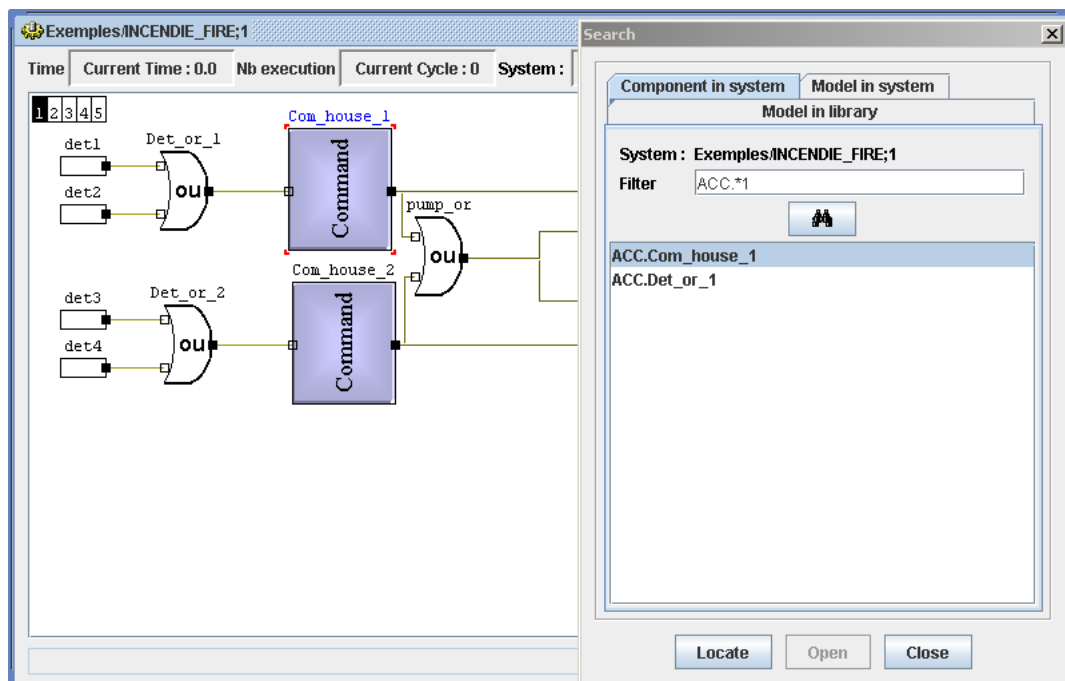


Figure 23. Search and selection of a component in simulation

## To take determinist events into account

### Temporized Transitions

## Introduction

In original AltaRica model, events are not classified in categories. In case of dysfunctional studies, you need to classify events in different categories.

Failures are random, they have to be considered as stochastic events. In order do that, we link event to a probability law (exponential, weibull, ...) It is done with external clause law of the model (See Standard manual of Altarica Extended language).

System re-configuration event that arise from failure, must be considered either as instantaneous event, or temporized one (must happen at the end of a  $\delta$  time) This type of event is defined with a Dirac probability law with  $\delta$  parameter.

```
extern law <event reconfig> = Dirac(3);
```

If  $\delta$  is equal to 0, event is instantaneous. If one of these transitions is fireable, it must be fired instantaneously.

If  $\delta$  is greater than 0, a transition which is fireable à t, will be fired after  $\delta$  time only if transition stays fireable between t and t+ $\delta$ .

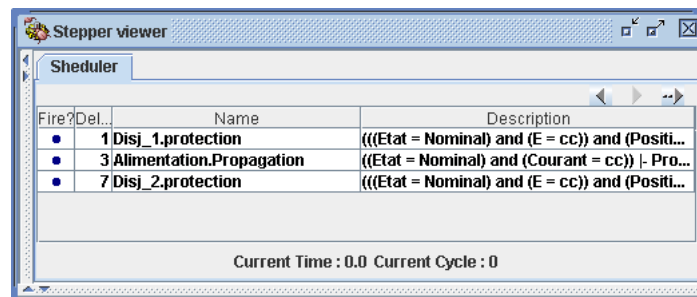
Temporised event is an extension of Altarica model which is not a reconsideration of overall behavior.

## Consideration in scheduler

If temporized events are taken into account, a scheduler must be managed in order to sort transitions functions of determinist delay related to events labelling transitions, and functions of dates when transitions become fireable.

There are only temporized transitions in scheduler. In fact, stochastic events can happen at any time, possibly between 2 temporized events.

Scheduler view displays temporized transitions that are valid. This list is sorted functions of ( $\Delta t$ ) fire delay related to each transition.



Fire?Del...	Name	Description
•	1 Disj_1.protection	((Etat = Nominal) and (E = cc)) and (Positi...
•	3 Alimentation.Propagation	((Etat = Nominal) and (Courant = cc)) - Pro...
•	7 Disj_2.protection	((Etat = Nominal) and (E = cc)) and (Positi...

Current Time : 0.0 Current Cycle : 0

When a temporized transition if fired, simulation current time is increased of  $\Delta t$ .

## Instantaneous transitions

Instantaneous transitions are often used to update informations inside AltaRica model, and for instantaneous reconfiguration of system. It reprints expected functionality of system. When model is OK, users don't want to see these transitions anymore, they want step by step simulator to take into account and hide them.

Simulator can be set-up in order to automatically fire instantaneous transitions. In this case, after each state change, simulator verifies that scheduler is not empty. If it is not, and if the first transition is instantaneous, simulator triggers this transition without asking user.

If simulator is set-up to automatically fire instantaneous transitions, it may be blocked in some case where there is always a fireable instantaneous transition in scheduler.

In order to stop this simulation, step by step simulator can display a message saying that too many instantaneous transitions have been fired sequentially.

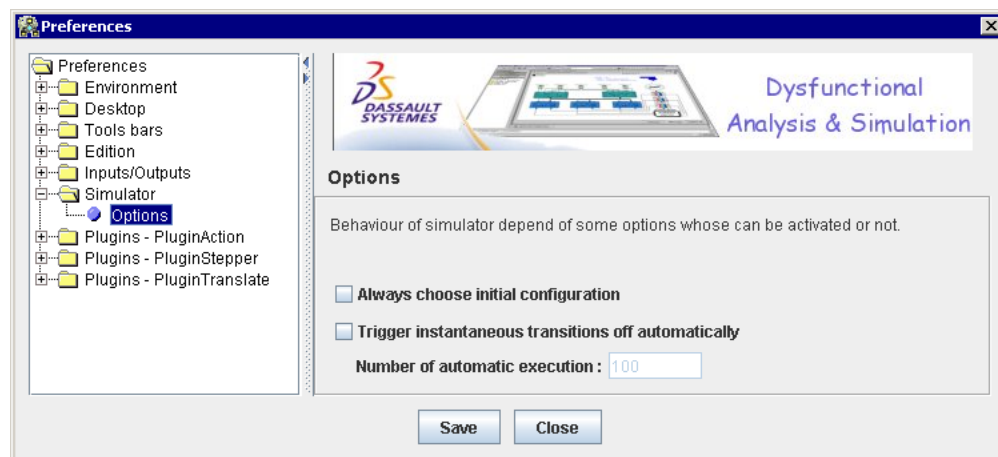
When an instantaneous transition is fired, a `cycle` number is increased, When a non-instantaneous transition is fired, the `cycle` number is reset to 0.

### Note

Scheduler view displays current time in bottom left corner, and current cycle in bottom right corner.

## Configuration of step by step simulator

Step by step simulator can be configured in software preferences. (=> **Options** menu, **Preferences** command, **Preferences/Simulation/Options** path)



## Temporal conflict between 2 transitions

A good question that users should ask is : *What happens when 2 temporized transitions are fireable at the same time ?*

There is no warranty of conflict management with step by step simulator. To help user to manage conflict, they can define a priority on model event. It can be made with `priority` external clause. (See Standard manual of Altairica Extended language)