

# **“SHARPE Interface” User's Manual**

## **Version 1.01**

Contact information:

Professor Kishor S. Trivedi

Center for Advanced Computing and Communication (CACC)

Department of Electrical and Computer Engineering

Duke University

Durham, NC 27708, USA

Phone: (919) - 660 - 5269

Fax: (919) - 660 - 5293

[kst@ee.duke.edu](mailto:kst@ee.duke.edu)

August, 1999

# Preface:

The purpose of this manual is:

- to introduce the SHARPE (Symbolic Hierarchical Automated Reliability and Performance Evaluator) modeling tool and illustrate its features.
- to introduce the Graphical User Interface SHARPE GUI.

SHARPE is a good tool to use for learning about and applying modeling techniques because it supports a variety of model types and, for most of the model types, provides more than one analysis algorithm from which the user can choose. It allows the user to choose when to combine different model types, which ones to combine, and how to combine them. SHARPE models were designed to answer the question: given time-dependent functions that describe the behavior of the components of a system and a description of the structure of the system, what is the behavior of the whole system as a function of time? The functions might be cumulative distribution functions (CDFs) for component failure times, CDFs for task completion times, or the probabilities that components are available at a given time. The system structure might be specified, for example, in the form of a fault tree, a task graph or a Markov chain.

The time-dependent functions describing the component behavior are restricted by SHARPE to be exponential polynomial in form. This is not a great restriction, because many of the most commonly used distribution functions have this form and much of the work that has been done in fitting distributions to data has used exponential polynomials.

The use of SHARPE in many universities and companies, the new requirements of the users, the textual interface of the package, were pushing developers to have a GUI of SHARPE in the market. This interface is implemented in JAVA to be portable in most of any platform. This version has 8 types of model included in it (Fault tree, Reliability block diagram, reliability graph, Markov chain, Product-form queuing network, Multi-chain Product-form queuing network, Series-Parallel graph, Generalized stochastic Petri net).

# Contents

Preface

## **1 Introduction 8**

1.1 Organization of this manual

1.2 Conventions used

## **2 Running SHARPE Interface 9**

## **3 Project 10**

3.1 Creation of a project

3.2 Save a project

3.3 Modification of the current project properties

3.4 Information of the current project

3.5 Open a project

3.6 Close a project

## **4 Model 14**

4.1 Creation of a model

4.2 Modification of model properties

4.3 Save a model

4.4 Delete a model

4.5 Methods

## **5 Design of a fault tree model**

**16**

5.1 Creation of the model

5.2 More explanations of the design

5.2.1 Top Event

5.2.2 Gate

5.2.3 Event

5.3 Use of the buttonBar

## **6 Design of a Markov chain model**

**21**

6.1 Creation of the model

6.2 Methods - Markov chain

6.2.1 Panel popup

6.2.2 Node popup

6.2.3 Arc popup

6.3 Use of the buttonBar

6.3.1 How to create a node?

6.3.2 How to move a node?

6.3.3 How to draw an arc?

6.3.4 How to modify an arc?

6.3.5 How to copy, paste or cut a part of the Markov chain model?

6.3.6 How to delete some objects from the model?

6.3.7 How to refresh the model?

6.3.8 How to complete the matrix?

<b>7 Design of a reliability block diagrams model</b>	<b>26</b>
7.1 Creation - Reliability block diagram	
7.2 Methods - Reliability block diagram	
7.2.1 Block	
7.2.2 Bridge block	
7.3 Use of the buttonBar	
<b>8 Design of a reliability graph model</b>	<b>30</b>
8.1 Use of the buttonBar	
8.1.1 How to create a node?	
8.1.2 How to move a node?	
8.1.3 How to draw an arc?	
8.1.4 How to modify an arc	
8.1.5 How to copy, paste or cut a part of the model?	
8.1.6 How to delete some objects from the model?	
8.1.7 How to refresh the model?	
<b>9 Design of a generalized stochastic Petri net model</b>	<b>32</b>
9.1 Creation - Generalized stochastic Petri net	
9.2 Methods - Generalized stochastic Petri nets	
9.2.1 Panel	
9.2.2 Place	
9.2.3 Transition	
9.2.4 Arc	

## **10 Design of a product form queuing network model 36**

10.1 Creation - Product-form queuing network

10.2 Methods - Product-form queuing network

10.2.1 Panel

10.2.2 Station

10.2.3 Arc

## **11 Design of a multi-chain product form queuing network model 39**

11.1 Creation of the model

11.1.1 How to create a station?

11.1.2 How to move a station?

11.1.3 How to draw an arc?

11.1.4 How to modify an arc

11.1.5 How to delete some objects from the model?

11.1.6 How to refresh the model?

## **12 Design of a series-parallel graph model 41**

12.1 Creation of the model (Use of the buttonBar)

12.1.1 How to create a node?

12.1.2 How to move a node?

12.1.3 How to draw an arc?

12.1.4 How to delete some objects from the model?

12.1.5 How to refresh the model?

## **13 Design of a hierarchical model** **43**

### 13.1 Hierarchy feature

#### 13.1.1 From a “fault tree” model

#### 13.1.2 From a “reliability block diagram” model

### 13.2 Hierarchy structure

### 13.3 Hierarchy dialog box

## **14 Definition of constants, variables and functions** **46**

### 14.1 Declaration of a symbol

#### 14.1.1 Creation of a “Constant”

#### 14.1.2 Creation of a “Variable”

#### 14.1.3 Creation of a “Function”

### 14.2 Declare in each model the parameters

#### 14.2.1 Fault tree - Parameter for an event

#### 14.2.2 Reliability block diagram - Parameter for a block

#### 14.2.3 Markov chain - Parameter for arc

#### 14.2.4 Generalized stochastic Petri net - Parameter for place/transition/arc

#### 14.2.5 Product-form queuing network - Parameter for station/arc

## **15 Analysis of a model** **50**

### 15.1 Analysis of the model

### 15.2 Parameter not defined in the main model, Outputs chosen

### 15.3 SHARPE code from the current model

### 15.4 SHARPE output from the current model

### 15.5 Graph

15.6 Personal modification of the SHARPE code

## **16 Plotting the results of a model** **54**

16.1 Open a plot

16.2 Combine a plot

16.3 Print a plot

16.4 Save a plot

16.4 Zoom the plot

## **17 Help features inside the GUI** **56**

17.1 Use of two statusBars:

17.2 Definition of the button Bars

17.3 Browse Frame

17.4 Help Frame



# Chapter 1

## Introduction

### 1.1 Organization of this manual

The guide assumes the user already has some experience with Reliability Availability. It is intended for first time users of the SHARPE interface.

### 1.2 Conventions used

The following conventions are followed in the manual

1. All buttons are displayed in boldface.
2. All pop-up buttons are displayed in italic.
3. A click is always left mouse button unless explicitly mentioned otherwise.
4. LMB: left mouse button; RMB: right mouse button.

# Chapter 2

## Running SHARPE Interface

The user should have received a zip disk containing the directory Sharpe-Gui or the file Sharpe-Gui.zip.

- 1/ To start the interface, you need to put the directory Sharpe-Gui from the zip disk in the directory C:\ (Sharpe-Gui contains the interface code) or unzip the file in the directory C:\
- 2/ Open Windows Explorer, then go in the directory C:\Sharpe-Gui, click on the icon "Interface SHARPE".

If you have in your MS-DOS prompt, the message "Out of the memory for the initial environment", follow these instructions: (For Windows95 & Windows98)  
At the top of this MS-DOS prompt, you have some icons. Click on the "Properties" icon. Then another window appear (Click on "Memory" in the tabset at the top of this window). You need to change the value of the memory allowed for the initial environment (top right). The value should be at this time "Auto", put it to 2048. Then click on the button "Apply" (bottom right in the window), then "Ok". Close the MS-DOS prompt and click again and the "Interface SHARPE" icon.

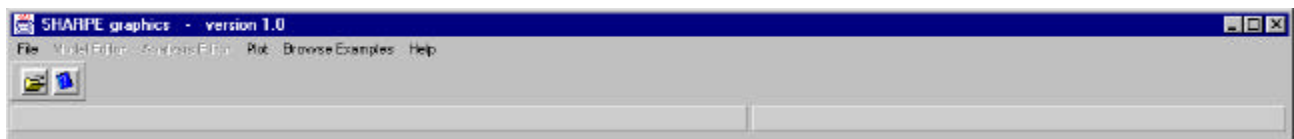


Figure 2.1: Main frame

When the application is started, the user can see a frame containing 6 menus (Fig 2.1):

- File
- Model Editor
- Analysis Editor
- Plot
- Browse Examples
- Help

Each of these menus will be described in the following chapters.

# Chapter 3

## Project

### 3.1 Creation of a project

The user should click on “File” menu then “New” menu to create a new project or use the shortcut reserved for this action.

Shortcut: Ctrl-N (anytime during the session)

1. When the “Information for the new project” dialog box appears, fill out the various fields and click “OK”.

The different fields:

- A project name (name.rgl will be the project file containing the information for all the models created).
- The type of the first model which will be implemented in this project.
- The name of the first model.
- The owner's name.
- The text area reserved for the description of the new project called “Commentary”.

2. A frame now appears on the screen after the validation of the user. The frame consists of 2 parts:

- The left part is a list containing the names of the models present in this project.
- The right part is the panel used to design the current model (selected in the list from the left part).

### 3.2 Save a project

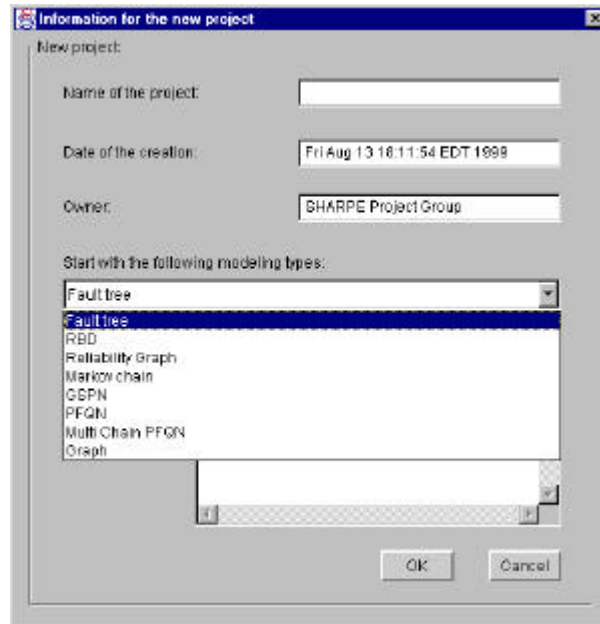
The user can save the current project with all the models implemented inside it. The user can also save a single model (see “Save a model”).

Projects are saved with the extension .rgl and the models are saved as:

- .flt (fault tree)
- .rbd (reliability block diagram)
- .mkv (Markov chain)

- .gpn (generalized stochastic Petri net)
- .pfq (product-form queuing network)
- .mpf (multi-chain product-form queuing network)
- rg (reliability graph)
- gph (series-parallel graph)

Figure 3.1: New project



### 3.3 Modification of the current project properties

Click on the “Model Editor” menu then on “Information”. The “Project and models information” dialog box should appear.

In this dialog box, the user can modify:

- The name of the current project
- The owner's name
- The description of the project

Click on the “Validate” button to update the values.

### 3.4 Information of the current project

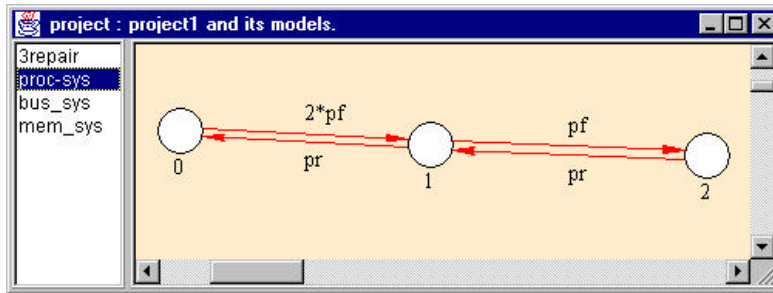
1. To change the name of the project:

Click on the “Model Editor” menu and then on the “Information” menu. Click on the “Project” tab. You can also change the owner's description and the commentary.

2. To change the name a model from the current project:

Click on the “Model Editor” menu and then on the “Information” menu. Click on the “Models” tab.

You can visualize the links between all the models created in the current project.



A tree displays the hierarchy structure of all the models.

Figure 3.2: Project property

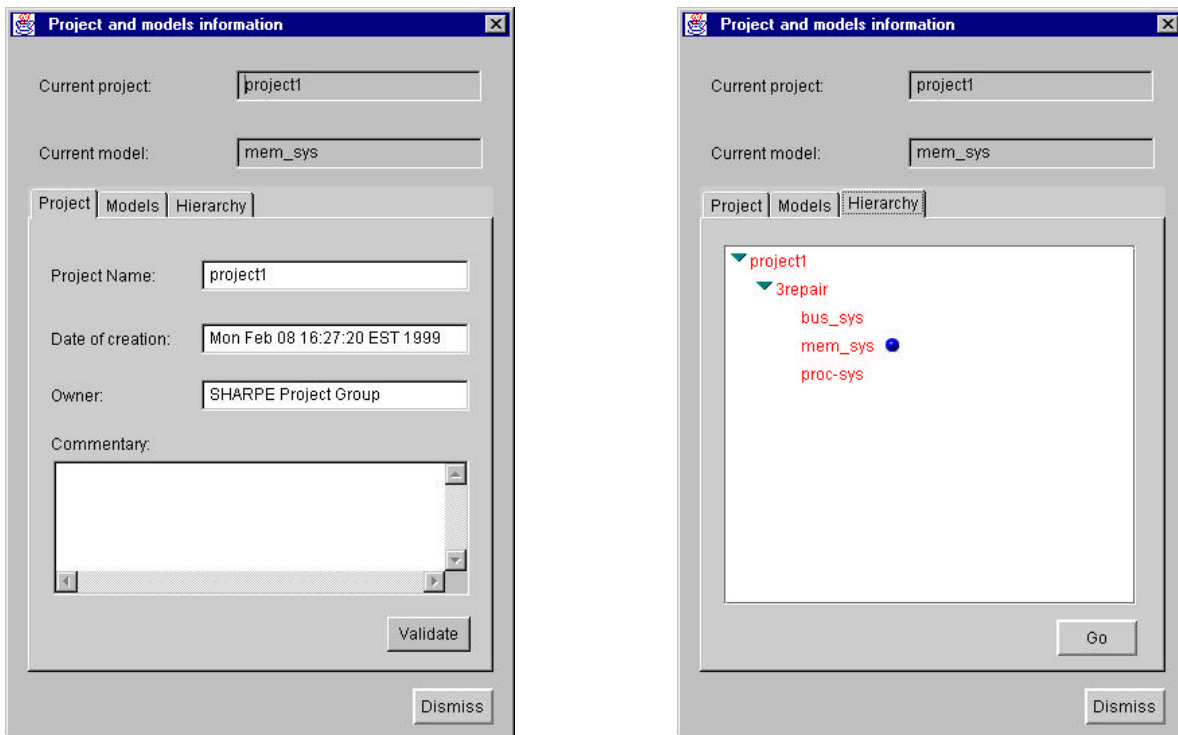
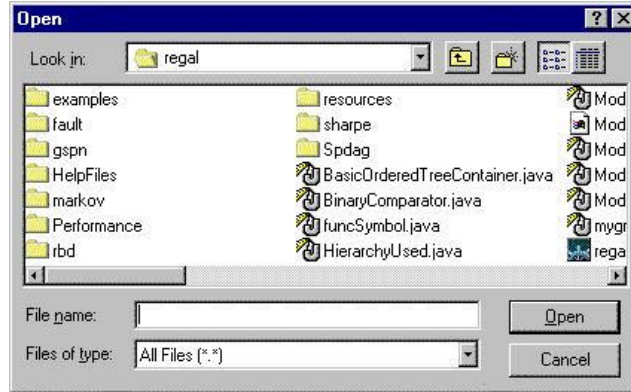


Figure 3.4: Tree containing the structure of the current project

### 3.5 Open a project

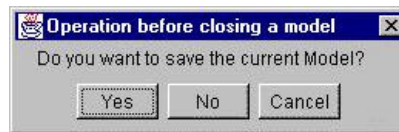
The user should click on “File” menu then “Open” menu to open a project

When the following dialog box appears, choose a .rg file and click “OK”. The extension of the project files in the interface is always .rgl



### 3.6 Close a project

When the user closes a project, this dialog appears asking to save before closing.



# Chapter 4

## Model

### 4.1 Creation of a model

When a new project is being created, the interface requests a name for the first model that will be implemented.

To add a model, click on the “Model Editor” menu and choose from the model types available.

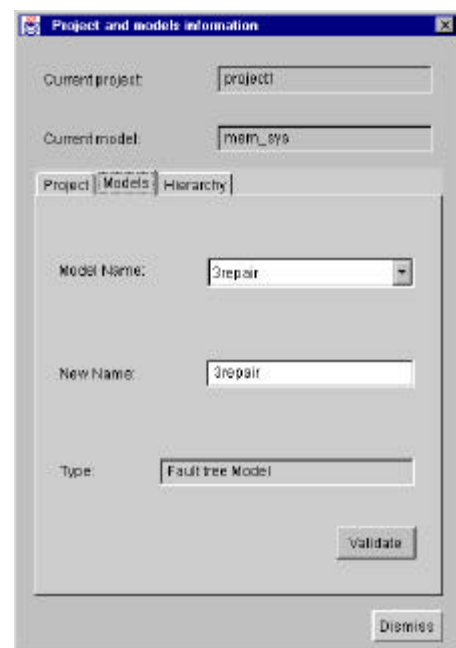
The user can create, using the “Hierarchy feature”, a new model called by a model previously implemented

### 4.2 Modification of model properties

Pull the “Model Editor” menu down and click on “Information” menu. The “Project and models information” dialog box appears.

After clicking on “Models” on the tabset, change the name of the selected model.

Click on the “Validate” button to update the information.



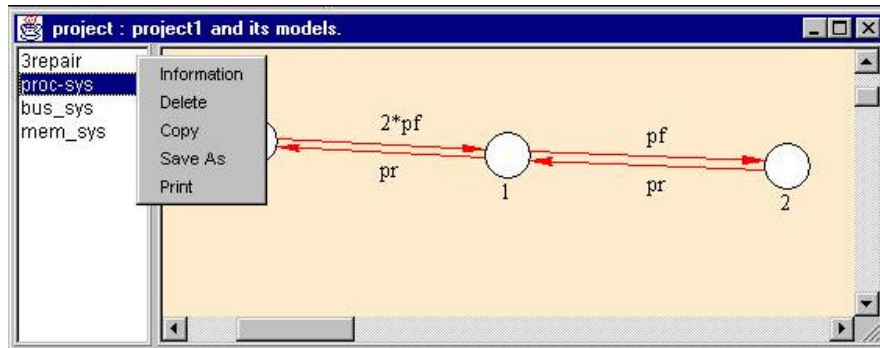
### 4.3 Save a model

The user has the possibility to save a model outside a project. Click on the “File” menu and select “Save Model”. Models are saved with the following extensions:

- .flt (fault tree)
- .rbd (reliability block diagram)
- .mkv (Markov chain)

- .gpn (generalized stochastic Petri net)
- .pfq (product-form queuing network)

Figure 4.2: Delete a model



#### 4.4 Delete a model

To delete a model from the current project, click with the right button on the model's name, which appears in the list inside the project's frame. This action is available only when 2 models or more have been created inside the current project.

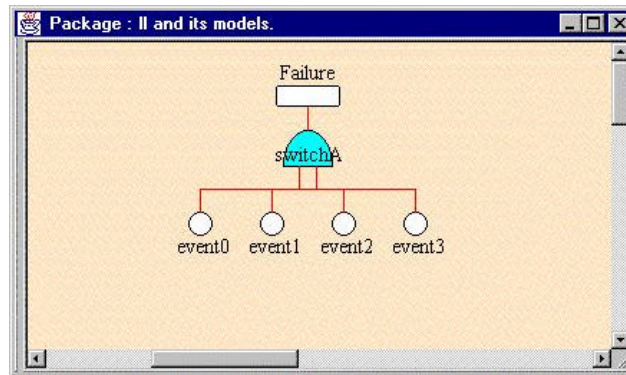
#### 4.5 Methods

Each type of model has its own way to be designed. When a model is created or opened, a canvas appears on the right side of the current package's frame. Clicks with the right button of the mouse will generate different pop-ups depending on the object that has been clicked on. In the chapters, a description of all the pop-ups possible is given and also the description of the buttonBar related to the type of the current model.



# Chapter 5

## Design of a fault tree model



### 5.1 Creation of the model

After selecting a “fault tree” model, the user should design his/her model by clicking on the “Top event” object with the right button of the mouse.

Right mouse button clicks generate popups.

- You can create a gate by clicking on an event then click on “Add gate”.
- You can change a gate to an event by clicking on a gate then click on “Change to an event”.
- You can enter the parameter of an event by clicking on an event then click on “Parameters”.

### 5.2 More explanations of the design

After the fault tree model is selected, there will be one block displayed in the panel. This block serves as the top event of the fault tree. The construction of the model begins by clicking on the top event block. A pop-up will then appear on the screen. After clicking on “Add Gate”, a dialog box will appear. Select one of the gate types and fill in the fields that appear.

Once the gates have been created, more gates can be added or the properties of the created events can be changed. Basic events, which are created initially, will have a

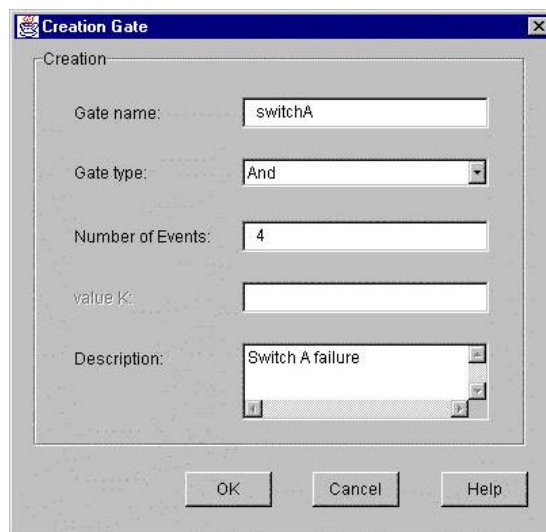
white circle to signify that the property for these events has not been defined. Once the property has been defined for a particular event, the event will be represented in orange. To define the properties for a particular event, select the desired event by clicking the right mouse button over the event.

A pop-up will appear on the screen. We will explain for each object, the methods available:

### 5.2.1 Top Event

The only possibility at the beginning of the creation is adding a new gate. When the model contains at least one gate, the pop-up contains only the “Clear” action.

Figure 5.1: Add a gate



#### “Add Gate” action

The “Add Gate” pop-up appears. Choose the gate to be created and also the number of input events the new gate will have. A description of the gate can be given at this time. After clicking on the “OK” button, the new gate is shown in the panel.

#### “Clear” action

The “Clear” action deletes all the objects, which are in the panel area.

### 5.2.2 Gate

#### “Add event” action

This is used to change the number of events input to the gate. Enter only a number lower to the previous value. To have fewer events than the current number, use the “Cut” action on the future deleted events.

Figure 5.2: Gate's pop-up



#### “Change to an event” action

This action is used to change a gate to an event.

#### “Copy” action

This action will copy the current gate to the clipboard: if an event is selected after copying a gate and “Paste” action selected from the pop-up, the event will be replaced by the copied gate. When a gate is copied from the canvas, all gates and events, which are inputs to that gate, are placed in the clipboard.

#### “Cut” action

This action will delete the selected gate. Clicking on an event and choosing

#### “Paste”

will result in the event being replaced by the cut gate. When a gate is cut from the panel, all gates and events, which are inputs to that gate, are deleted and placed on the clipboard as well.

#### “Identify” action

The Identify button will display a frame contained the description of the current gate. When the mouse enters another gate, the frame displays the description of this gate. The user should close the windows by clicking on “Identify” again or on the “X” (top-right) of the frame.

### 5.2.3 Event

#### “Parameters” action

A frame appears on the screen to help to define each event of the current fault tree. (See parameter creation).

Figure 5.3: Event's pop-up



#### “Add Gate” action

The “Add Gate” pop-up appears. Choose the gate to be created and also the number of input events the new gate will have. A description of the gate can be given at this time. After clicking on the “OK” button, the new gate is shown in the panel.

“Copy” action

The “Copy” button will copy the selected parameter of the current event and put it on the clipboard for a subsequent paste operation on another event.

“Cut” action

The Cut button will delete the selected object from the panel and put it on the clipboard for a subsequent paste operation.

“Paste” action

The Paste button will take the object(s) currently in the clipboard and paste it in the location of the selected event. This event will be replaced by the object(s) that are in the clipboard.

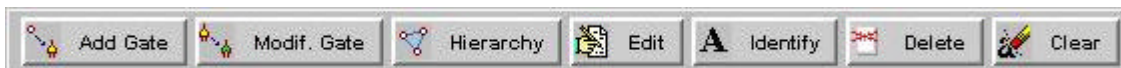
“Hierarchy” action

(See “Hierarchy called”)

“Identify” action

The Identify button will display a frame contained the description of the current event. When the mouse enters another event, the frame displays the description of this event. The user should close the windows by clicking on “Identify” again or on the “X” (top-right) of the frame.

### 5.3 Use of the buttonBar



- Add Gate:  
Click on “Add Gate” then click on an event to replace it by a new gate.
- Modif. Gate  
Click on “Modif. Gate” then click on a gate to modify this gate (can modify the type (AND,OR,KOFN), can modify the number of events, enter a description).
- Hierarchy  
Click on “Hierarchy” then on an event to have this event calling another model.
- Edit  
Click on “Edit”, then the Parameter’s frame to initialize each event appears.
- Identify  
Click on “Identify” then put the mouse on the top of an event,  
The user is able to type a description of the selected event.

- Delete  
Click on “Delete” then click on an event or a gate which would be deleted.
- Clear  
Click on “Clear” will erase all the objects in the panel.

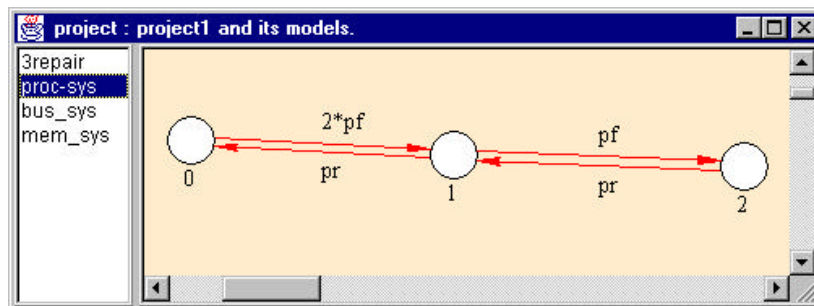
# Chapter 6

## Design of a Markov chain model

### 6.1 Creation of the model

After selecting a “Markov chain” model, the user should design his/her model by clicking on the panel with the right button of the mouse.

Right mouse button clicks generate popups.



### 6.2 Methods - Markov chain

When the user clicks on an object, a pop-up appears on the screen. We will explain for each object, the methods available:

#### 6.2.1 Panel popup

Figure 6.1: Pop-up after clicking on the Markov model's panel

“Type Markov Chain” action



At any time during the design the type of the current Markov chain can be known. The possible types are:

- Absorbing Markov chain
- Irreducible Markov chain
- Non-irreducible Markov chain
- Multi-chain (many independent Markov chains)
- Markov chain with isolated node(s)

## 6.2.2 Node popup

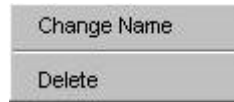


Figure 6.2: Pop-up after clicking on a node

### “Change Name” action

Used to change the name of a node.

### “Delete” action

When a node is removed, all the arcs, which were connected to it, will be removed also.

## 6.2.3 Arc popup

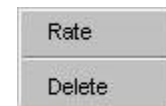


Figure 6.3: Pop-up after clicking on an arc

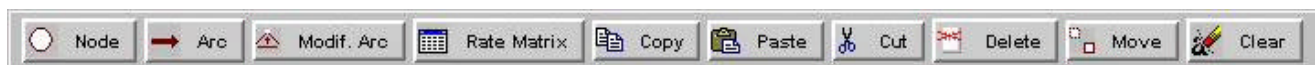
### “Rate” action

A value for the rate of an arc may be entered with the “Panel/Rate matrix” action, or through this action. After entering and validating the new value, this value will be displayed inside the panel, in the middle of the arc selected.

### “Delete” action

This removes an arc.

## 6.3 Use of the buttonBar



### 6.3.1 How to create a node?

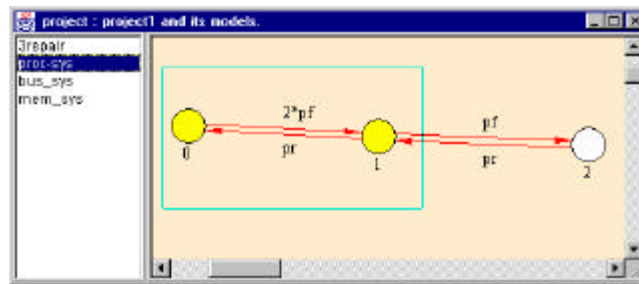
Click on “Node” then click on the panel with the left button of the mouse..

A dialog box appears asking for the name of the new node, click on “OK” to create the node.

### 6.3.2 How to move a node?

Click on “Move” in the buttonBar then click on a node and drag the mouse to the new location, finally release the mouse.

If you want to move many nodes at the same time, you can select many nodes neither by clicking one after one with the left button and Ctrl key pressed or draw a rectangle containing the expected nodes (left click on the panel then drag the mouse and release it). When many nodes are selected, move one of the nodes selected will move all of them.

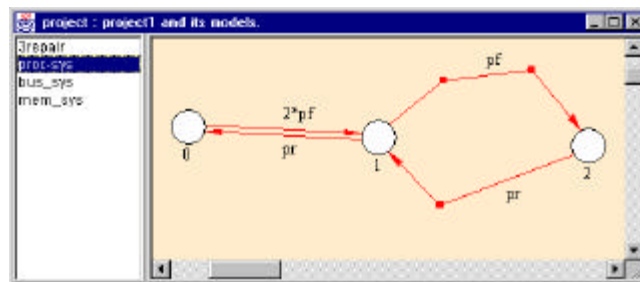


### 6.3.3 How to draw an arc?

Click on “Arc” in the buttonBar then click on a node and drag the mouse. By releasing the mouse over another node, the arc is draw connecting the 2 nodes.

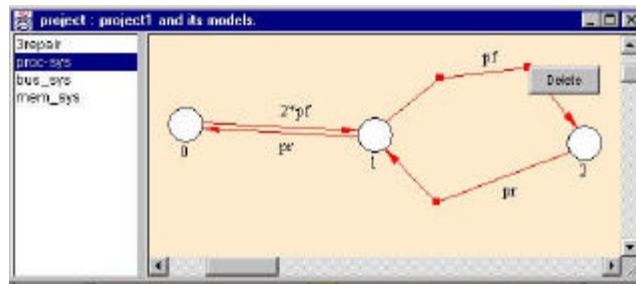
The release of the button, inside the panel (and not on a node), will cancel the action of drawing the arc.

### 6.3.4 How to modify an arc?



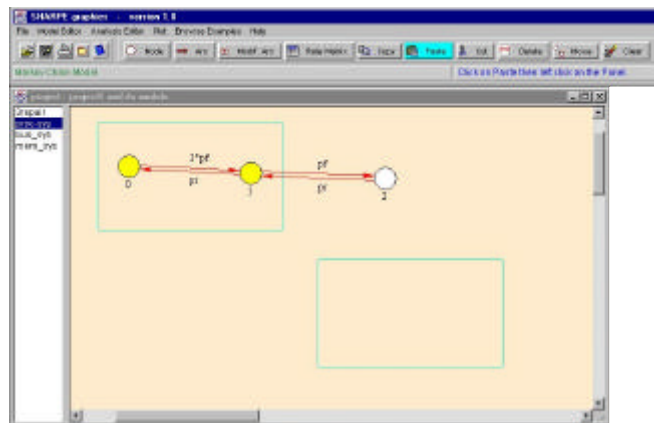
Click on “Modif. Arc”, the intermediate points are then representing by a small square. You can move the squares by clicking (left button) over a square then drag the mouse until the new location. The release of the mouse will put the selected square to the location chosen. You can add intermediate points by clicking on an arc and drag the mouse to the location expected for the new point. You can also delete any intermediate point by clicking with the right button of the mouse over a square and select the action “Delete”.





### 6.3.5 How to copy, paste or cut a part of the Markov chain model?

Select many nodes neither by clicking one after one with the left button and Ctrl key pressed or draw a rectangle containing the expected nodes (left click on the panel then drag the mouse and release it). When many nodes are selected, you can click on “Copy” or “Cut”. By clicking on “Paste”, and moving the mouse over the panel, another rectangle appears which contains the new nodes created. The user should click on the new location to display the new nodes. The “Paste” action is available only after the user has selected nodes and clicked on “Copy” or “Cut”



### 6.3.6 How to delete some objects from the model?

Click on “Delete” then click on a node or an arc which would be deleted

### 6.3.7 How to refresh the model?

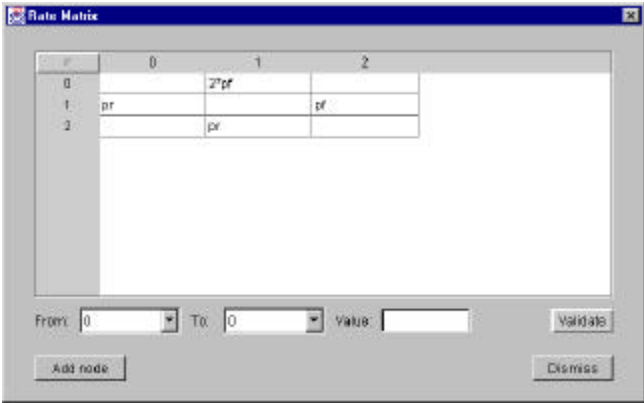
Click on “Refresh” to repaint the panel.

### 6.3.8 How to complete the matrix?

“Rate Matrix” action

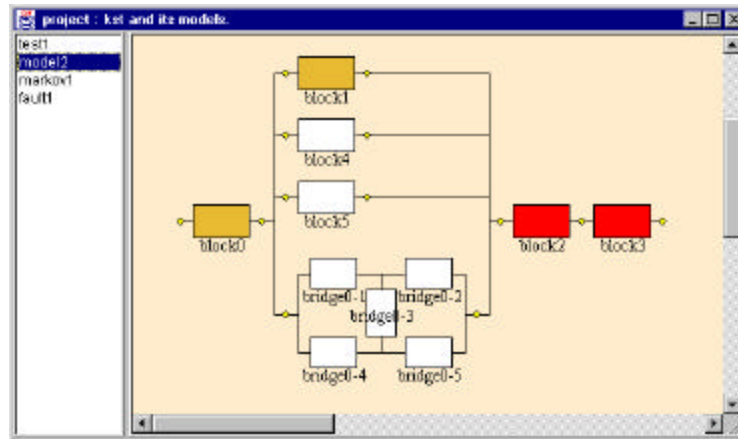
This is used to enter the rate values of each of the already created arc, as well as for arcs that have not been yet implemented. The yellow rectangles show the arcs already implemented but which don't have a rate value at the current time. Values entered in each of the white rectangles inside the grid will generate new arcs in the panel after the dialog box has been exited. The button “add” creates a new node (state) and enters a new column and row inside the grid. A new node will also appear on the screen after this action. Because it is difficult to know where to insert the new node in a good location, the node will be placed on the top left of the panel. If more than one node is created with the button “add”, the new nodes will be located in the same spot. Each new node can be moved to the desired position after leaving the dialog box (clicking the button “OK”).

Figure 6.4: Rate matrix



# Chapter 7

## Design of a reliability block diagrams model



### 7.1 Creation - Reliability block diagram

After selecting a “RBD” model, the user should design his/her model by clicking on the only block at this time present in the panel with the right button of the mouse. Right mouse button clicks generate popups. The popup sections are explained in the “Methods Actions” part.

For examples:

- You can create a series structure by clicking on a block then click on “Series”.
- You can create a parallel structure by clicking on a block then click on ”Parallel”.
- You can enter the parameter of a block by clicking on an event then click on “Parameters”.

### 7.2 Methods - Reliability block diagram

When an object is clicked on, a pop-up appears on the screen. We will explain for each object, the methods available:

#### 7.2.1 Block

“Parameter” action

A frame appears on the screen to help to define each block of the current rbd model. (See parameter creation).



### “Identify” action

The Identify button will display a frame contained the description of the current block. When the mouse enters another block, the frame displayed the description of this block. The user should close the windows by clicking on “Identify” again or on the “X” (top-right) of the frame.

Figure 7.1: Creation of a series structure

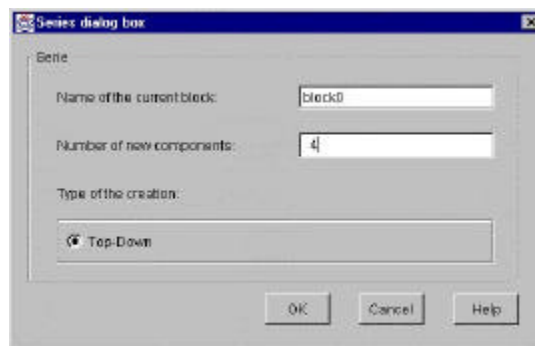
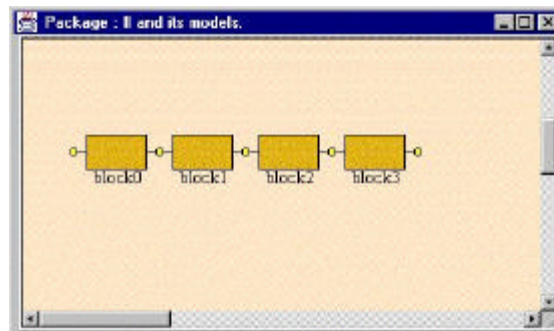


Figure 7.2: Creation of a series structure: result



### “Series” action

Creates a “series” structure.

### “Parallel” action

Creates a “parallel” structure.

Figure 7.3: Creation of a parallel structure

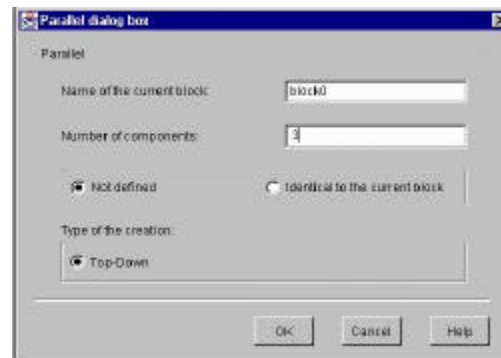
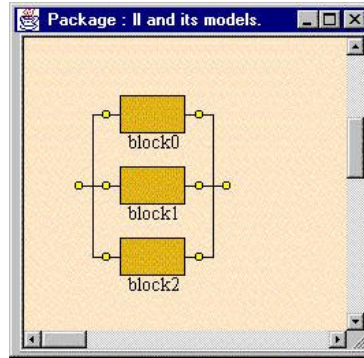


Figure 7.4: Creation of a parallel structure: result



“K out of N” action

Action not yet implemented. This is used to create a “K out of N” structure when some blocks contained in the structure are different. If all the blocks have the same component type, click on “Parameter” in the event's pop-up, then define the selected block to have its value “K” and “N” different from 1 (see parameters creation).

“Bridge” action

Creates a “bridge” structure.

“Delete” action

The Delete button will delete the selected object from the panel.

“Clear” action

The “Clear” button deletes all the objects, which are in the panel area.

“Undo” action

The “Undo” button erases the last action made in the design.

“Hierarchy” action

(See “Hierarchy called”)

### 7.2.2 Bridge block

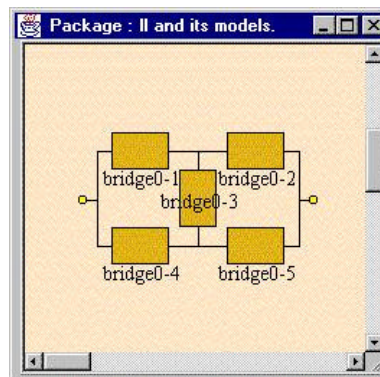


Figure 7.5: Pop-up after clicking on a block from a bridge



### “Parameter” action

A frame appears on the screen to help define each block of the current rbd model. (See parameter creation).

### “Delete bridge” action

The Delete button will delete the bridge contained the selected block from the panel.

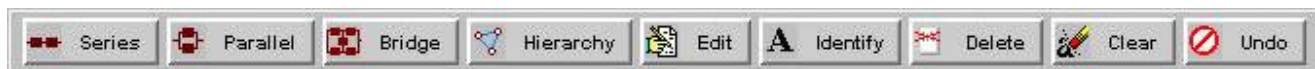
### “Clear” action

The “Clear” button deletes all the objects, which are in the panel area.

### “Hierarchy” action

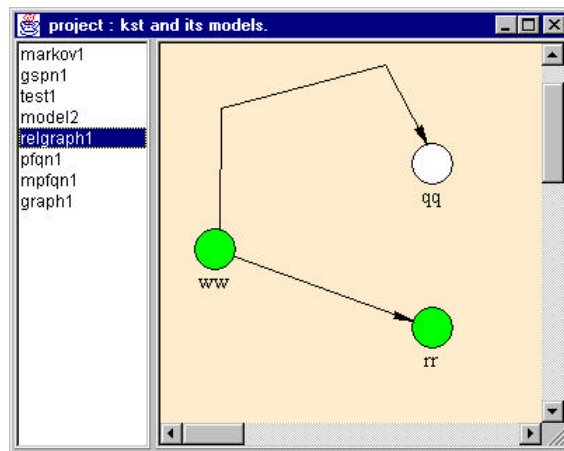
(See “Hierarchy called”)

## 7.3 Use of the buttonBar

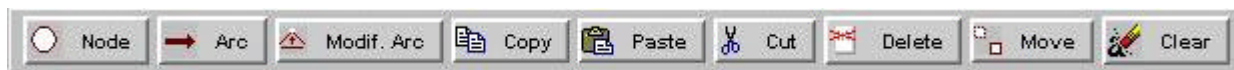


# Chapter 8

## Design of a reliability graph model



### 8.1 Use of the buttonBar



#### 8.1.1 How to create a node?

Click on “Node” then click on the panel with the left button of the mouse..

#### 8.1.2 How to move a node?

Click on “Move” in the buttonBar then click on a node and drag the mouse to the new location, finally release the mouse.

If you want to move many nodes at the same time, you can select many nodes neither by clicking one after one with the left button and Ctrl key pressed or draw a rectangle containing the expected nodes (left click on the panel then drag the mouse and release it). When many nodes are selected, move one of the nodes selected will move all of them.

#### 8.1.3 How to draw an arc?

Click on “Arc” in the buttonBar then click on a node and drag the mouse. By releasing the mouse over another node, the arc is draw connecting the 2 nodes.  
The release of the button, inside the panel (and not on a node), will cancel the action of drawing the arc.

#### 8.1.4 How to modify an arc

Click on “Modif. Arc”, the intermediate points are then representing by a small square. You can move the squares by clicking (left button) over a square then drag the mouse until the new location. The release of the mouse will put the selected square to the location chosen. You can add intermediate points by clicking on an arc and drag the mouse to the location expected for the new point. You can also delete any intermediate point by clicking with the right button of the mouse over a square and select the action “Delete”.

#### 8.1.5 How to copy, paste or cut a part of the model?

Select many nodes neither by clicking one after one with the left button and Ctrl key pressed or draw a rectangle containing the expected nodes (left click on the panel then drag the mouse and release it). When many nodes are selected, you can click on “Copy” or “Cut”. By clicking on “Paste”, and moving the mouse over the panel, another rectangle appears which contains the new nodes created. The user should click on the new location to display the new nodes. The “Paste” action is available only after the user has selected nodes and clicked on “Copy” or “Cut”

#### 8.1.6 How to delete some objects from the model?

Click on “Delete” then click on a node or an arc which would be deleted

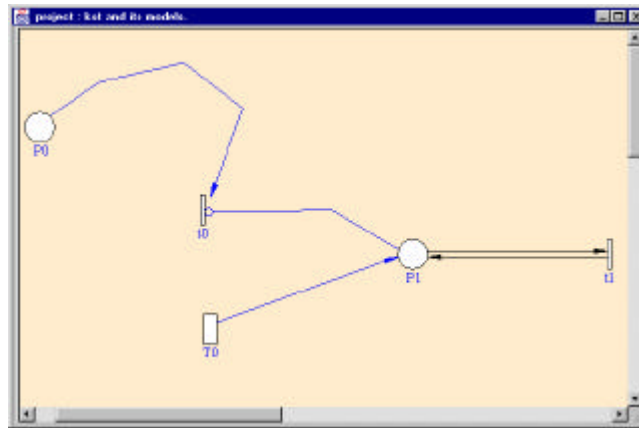
#### 8.1.7 How to refresh the model?

Click on “Refresh” to repaint the panel.



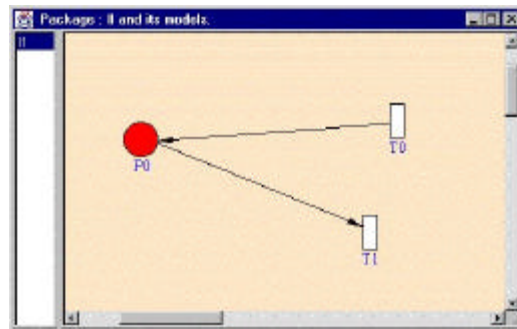
# Chapter 9

## Design of a generalized stochastic Petri net model



### 9.1 Creation - Generalized stochastic Petri net

After selecting a “GSPN” model, the user should design his/her model by clicking on the panel with the right button of the mouse. The principle is to click an object with the right button of the mouse then select the action from the popup. All the methods are explained in the “Methods Actions” part.



### 9.2 Methods - Generalized stochastic Petri nets

When the user clicks on an object, a pop-up appears on the screen. We will explain for each object, the methods available:

## 9.2.1 Panel

Clicking on left mouse button will create a place or a transition. To change this option, click on the buttonBar on “Place”, “Imm. Trans” or “Timed Trans.”

### “Clear” action

All the objects that are inside the panel will be deleted (places, transitions, arcs, and labels).

### “Undo” action

This action is not valid in this version of the interface.

## 9.2.2 Place

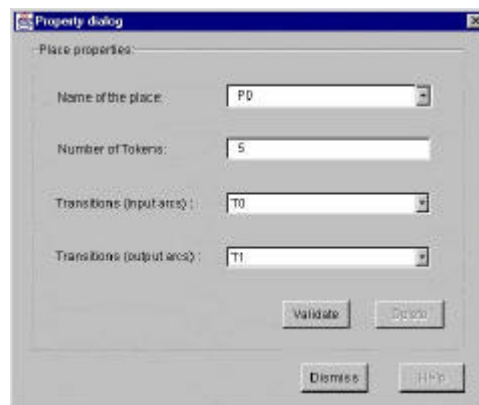
Figure 9.1: Pop-up after clicking on a place



### “Property” action

Used to assign a name and a number of tokens to the place.

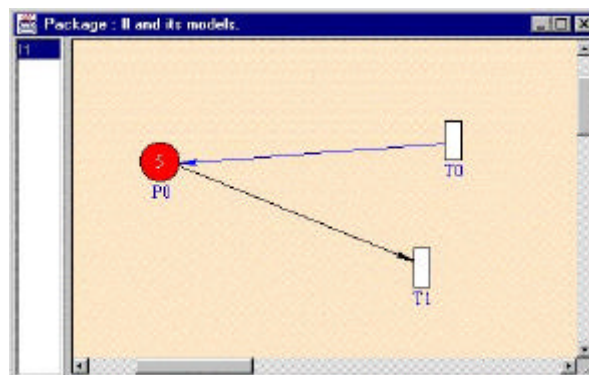
Figure 9.2: Property of a place



### “Display Number Token(s)” action

Used to display or not the number of tokens in the middle of all the places. If there is no token in one place, nothing is displayed instead to show the value 0.

Figure 9.3: Display the number of tokens



### “Hierarchy” action

This action is not valid in this version; it will be available in the next version.

### “Delete” action

When a place is removed, all the arcs that are connected to it will be removed also.

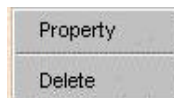
### How to move a place?

The user should click first on the button “Move” from the buttonBar then move a place by clicking on the object, then drag the mouse to the new location.

### How to draw arcs?

Arcs are drawn by clicking (left button of the mouse) on a place after selecting the button “Arc” or “Inhibitor Arc” from the buttonBar, then dragging the mouse, by releasing it over a transition, which should be connected by the new arc. The release of the button, inside the panel (and not on an object), will cancel the action of drawing the arc. Arcs between 2 places are not allowed.

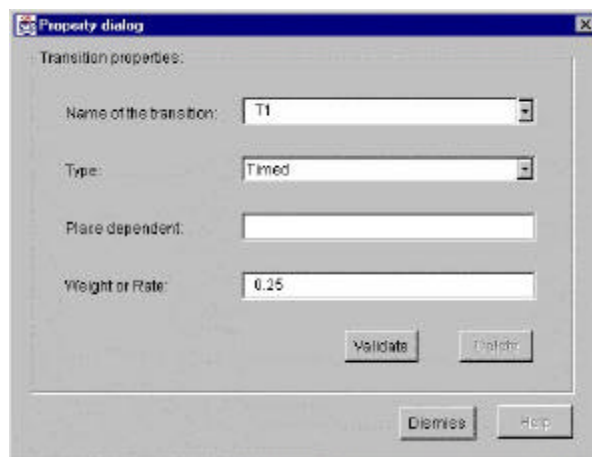
## 9.2.3 Transition



### “Property” action

Used to enter/modify the parameters (Name, Type of the selected transition, Weight or Rate (depending to the type of the transition), Place dependent) for the transition selected. Click “Validate” after changes are made. The dialog box can be exited by clicking on the button “Dismiss”.

Figure 9.4: Property of a transition



### “Delete” action

When a transition is removed, all the arcs, which were connected to it, will be removed also.

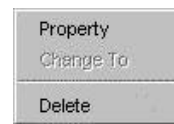
How to move a transition?

The user should click first on the button “Move” from the buttonBar then move the transition by clicking on the transition, then drag the mouse to the new location.

How to draw arcs?

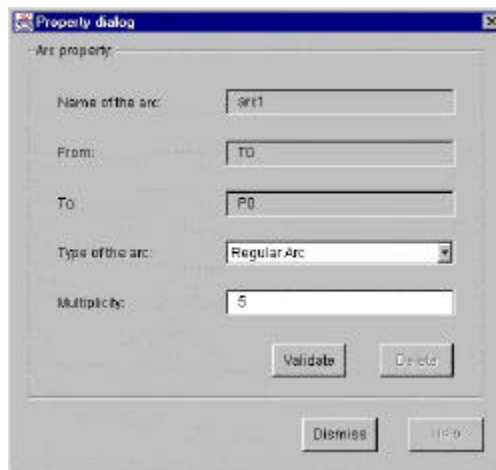
Arcs are drawn by clicking (left button of the mouse) on a transition after selecting the button “Arc from the buttonBar, then dragging the mouse, by releasing it over a transition, which should be connected by the new arc. The release of the button, inside the panel (and not on an object), will cancel the action of drawing the arc. Arcs between 2 transitions are not allowed.

Figure 9.5: Pop-up after clicking on an arc



### 9.2.4 Arc “Property” action

Figure 9.6: Property of an arc



The window appears on the screen. The user can modify the multiplicity value, but also can see the place and transition connected by it.

“Change To” action

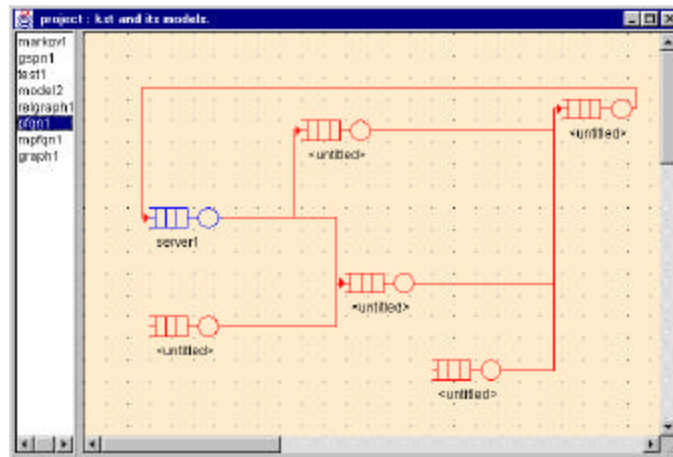
This action is not valid in this version of the interface.

“Delete” action

The current arc (receiving the click event) will be deleted when the user pressed the button “Delete” in the Arc pop-up.

# Chapter 10

## Design of a product form queuing network model



### 10.1 Creation - Product-form queuing network

After selecting a “PFQN” model, the user should design his/her model by clicking on the panel with the right button of the mouse.

Right mouse button clicks generate popups. The popup sections are explained in the “Methods Actions” part.

### 10.2 Methods - Product-form queuing network

When the user clicks on an object, a pop-up appears on the screen. We will explain for each object, the methods available:

#### 10.2.1 Panel

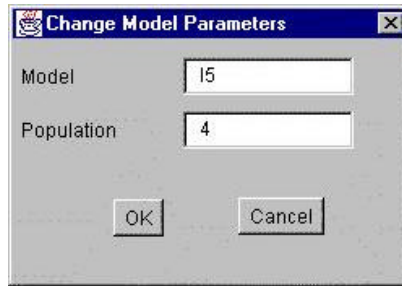
##### “Station” action

Stations are created by clicking on the panel with the left mouse button.

##### “Clear” action

The “Clear” button deletes all the objects, which are in the panel area.

Figure 10.1: Parameter for a PFQN model



“Refresh” action

The “Refresh” button will repaint the panel.

“Parameters” action

This is used to enter the model name and populations of the model.

### 10.2.2 Station

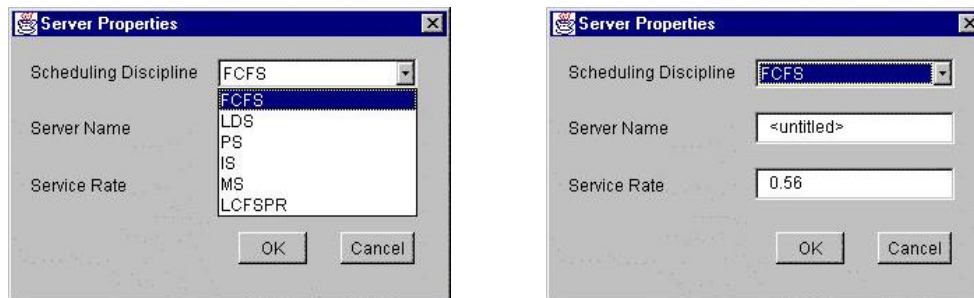
“Delete” action

This is used to delete a station. When a station is removed, all the arcs, which were connected to it, will also be removed.

“Properties” action

Used to assign name, service rate and distribution parameters for the station.

Figure 10.2: Property for a station



How to move a station?

Station can be moved by clicking on the node then dragging the mouse to the new location.

How to draw arcs?

Arcs are drawn by clicking (left button of the mouse) on a station, then dragging the mouse, by releasing it over another station, which should be connected by the new arc. The release of the button, inside the panel (and not on a station), will cancel the action of drawing the arc.

### 10.2.3 Arc



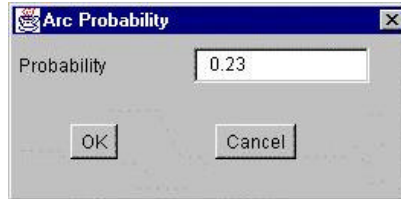
Figure 10.3: Pop-up after clicking on an arc

“Delete” action

The user can remove an arc by clicking on this button.

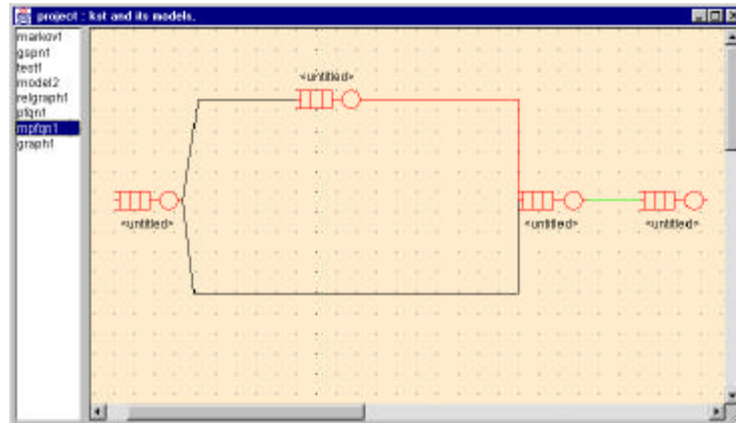
“Properties” action

Used to assign a probability to the arc.



# Chapter 11

## Design of a multi-chain product form queuing network model



### 11.1 Creation of the model



#### 11.1.1 How to create a station?

Click on “Station” then click on the panel with the left button of the mouse..

#### 11.1.2 How to move a station?

Click on “Move” in the buttonBar then click on a node and drag the mouse to the new location, finally release the mouse.

#### 11.1.3 How to draw an arc?

Click on “Arc” in the buttonBar then click on a station and drag the mouse. By releasing the mouse over another station, the arc is draw connecting the 2 nodes. The release of the button, inside the panel (and not on a node), will cancel the action of drawing the arc.

#### 11.1.4 How to modify an arc



Click on “Modif. Arc”, the intermediate points are then representing by a small square. You can move the squares by clicking (left button) over a square then drag the mouse until the new location. The release of the mouse will put the selected square to the location chosen. You can add intermediate points by clicking on an arc and drag the mouse to the location expected for the new point. You can also delete any intermediate point by clicking with the right button of the mouse over a square and select the action “Delete”.

#### 11.1.5 How to delete some objects from the model?

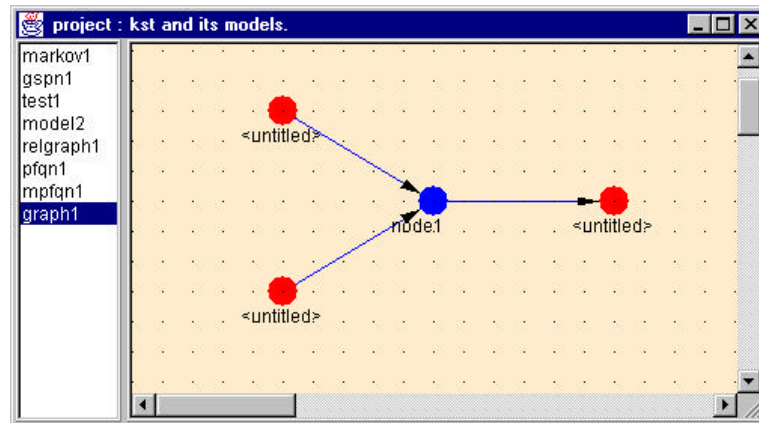
Click on “Delete” then click on a station or an arc which would be deleted

#### 11.1.6 How to refresh the model?

Click on “Refresh” to repaint the panel.

# Chapter 12

## Design of a series-parallel graph model



### 12.1 Creation of the model (Use of the buttonBar)



#### 12.1.1 How to create a node?

Click on “Node” then click on the panel with the left button of the mouse..

#### 12.1.2 How to move a node?

Click on “Move” in the buttonBar then click on a node and drag the mouse to the new location, finally release the mouse.

#### 12.1.3 How to draw an arc?

Click on “Arc” in the buttonBar then click on a node and drag the mouse. By releasing the mouse over another node, the arc is draw connecting the 2 nodes.

The release of the button, inside the panel (and not on a node), will cancel the action of drawing the arc.

#### 12.1.4 How to delete some objects from the model?

Click on “Delete” then click on a node or an arc which would be deleted

### 12.1.5 How to refresh the model?

Click on “Refresh” to repaint the panel.

# Chapter 13

## Design of a hierarchical model

### 13.1 Hierarchy feature

So far, we have shown how to use non-state-space models (reliability block diagrams, fault trees, and product-form queuing networks), Markov and GSPN models for performance and reliability analysis. The advantage of using non-state-space models (like block diagrams and fault trees) is that they are efficient to specify and solve.

However, the solution of these models assumes the components are independent. For instance, in a block diagram, fault-tree or reliability graph, the components must be completely independent of one another in their failure and repair behavior.

A failure in one component cannot affect the operation of another component, and components cannot share a repair facility. Markov models provide the ability to model systems that violate the assumptions made by the non-state-space models, but at the price of a state space explosion. A system having  $n$  components may require up to  $2^n$  states in its Markov chain representation: This problem can be handled in two ways:

It can be tolerated or it can be avoided:

Large model tolerance must apply to specification, storage and solution of the model. If the storage and solution problems can be solved, the specification problem can be solved by using more concise (and smaller) model specifications that can be automatically transformed into Markov models.

Large models can be avoided by using hierarchical model composition. The ability of SHARPE to combine results from different kinds of models makes it possible to use state-space methods for those parts of a system that require them, and use non-state-space methods for the more “well-behaved” parts of the system.

#### 13.1.1 From a “fault tree” model

Click on an event with the right button then click on “Hierarchy” in the popup. A frame is displayed. You can call a model, which is already in the current package, or call a new model. For a new model, you need to give the type and the output used for the SHARPE code.

See the paragraph “Hierarchy dialog box”.

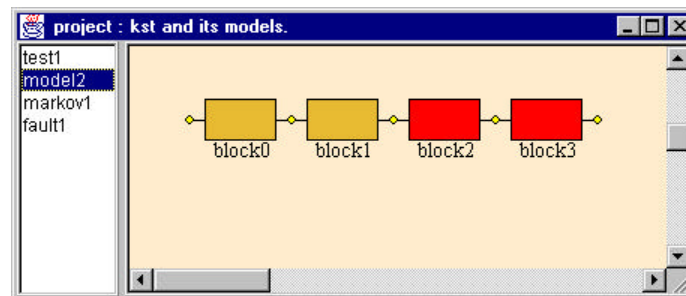
A red square represents an event, which called another model.



### 13.1.2 From a “reliability block diagram” model

Click on a block with the right button then click on “Hierarchy” in the popup. A frame is displayed. You can call a model, which is already in the current package, or call a new model. For a new model, you need to give the type and the output used for the SHARPE code.

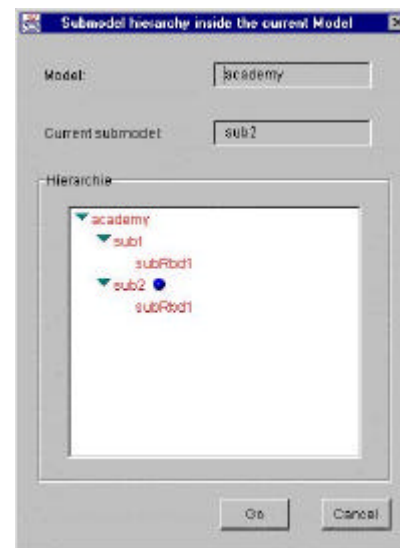
(See the paragraph “Hierarchy dialog box”).



A block, which called another model, is represented by a red rectangle.

## 13.2 Hierarchy structure

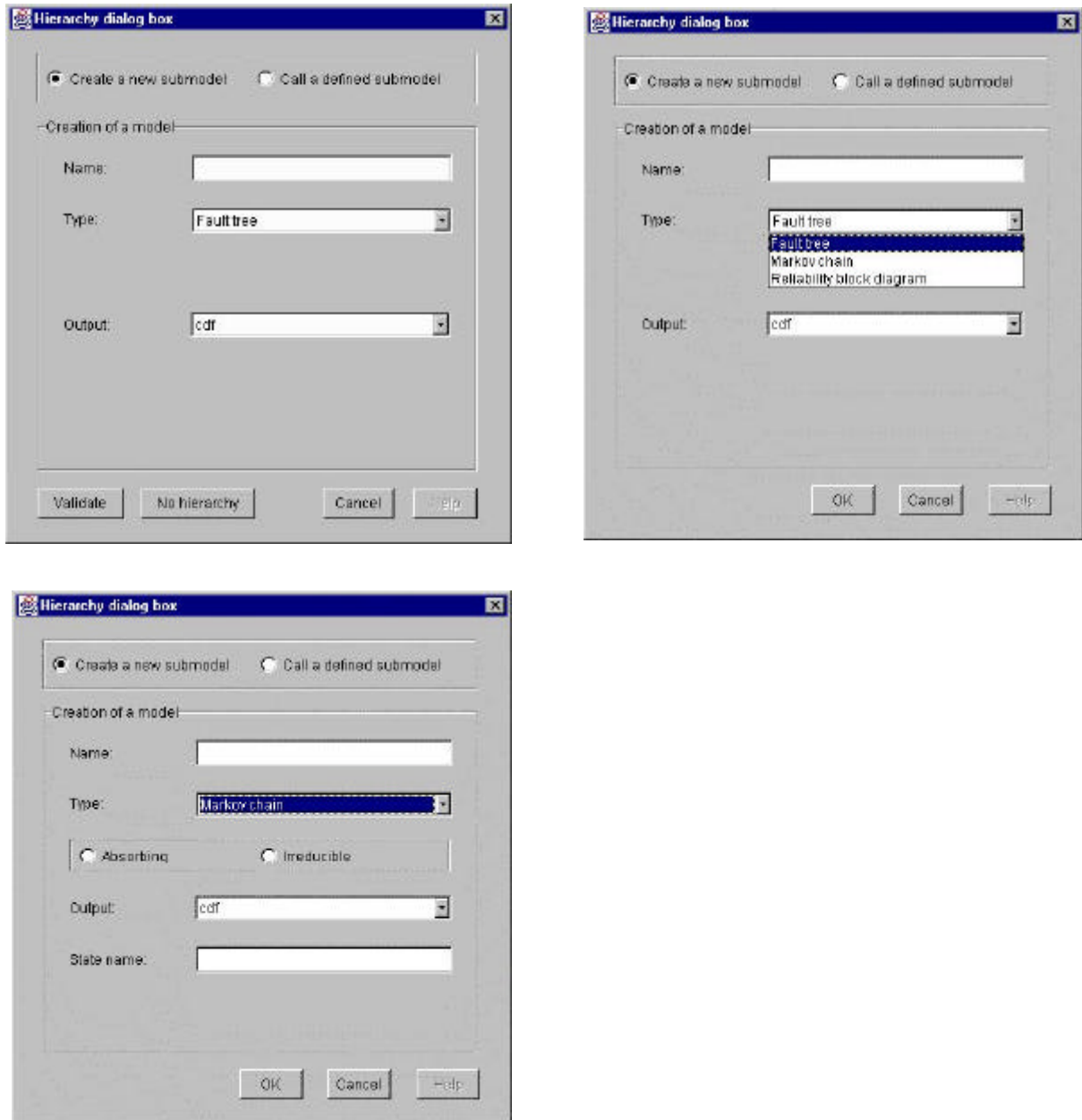
To visualize the hierarchy structure's tree between all the models created in the current package, click on the “Model Editor” menu and then on the “Information” menu. Click on the “Hierarchy” tab.



### 13.3 Hierarchy dialog box

Click on “Validate” to confirm the call from the current event to the specific model.  
Click on “No Hierarchy” to erase the link from the current event to a specific model.  
Click on “Cancel” to exit this dialog box without any changes.

Figure 13.1: Hierarchy frame



# Chapter 14

## Definition of constants, variables and functions

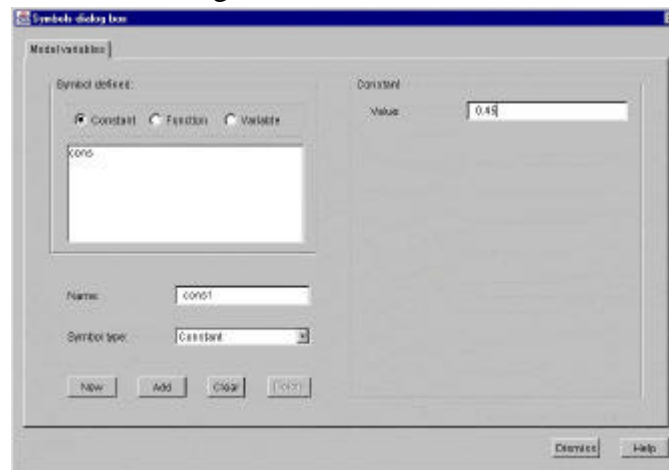
### 14.1 Declaration of a symbol

To create a new symbol, click on “Model Editor” menu and then on “Symbol” menu.

#### 14.1.1 Creation of a “Constant”

To create a new constant, click on the “New” button. Fill up the field Name then choose Constant for the field “Symbol type”. Enter a value. Click on “Add” to validate the creation. You can modify by clicking on the list (top left) after selecting Constant in the Symbol defined. The selected constant appears with its value on the screen, change the fields then click “Modify”.

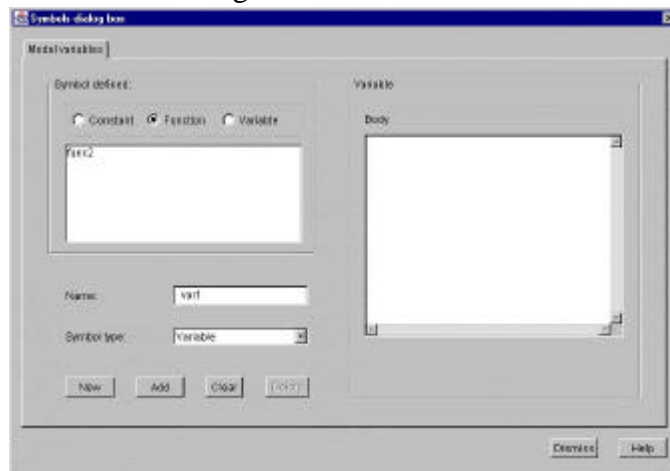
Figure 14.1: Constant



#### 14.1.2 Creation of a “Variable”

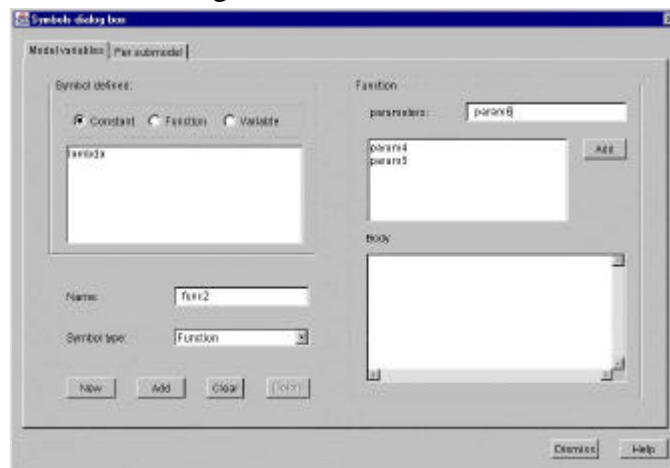
To create a new variable, click on the “New” button. Fill up the field Name then choose Variable for the field “Symbol type”. Enter the body of the new variable. Click on “Add” to validate the creation. You can modify by clicking on the list (top left) after selecting Variable in the Symbol defined. The selected variable appears with its value on the screen, change the fields then click “Modify”.

Figure 14.2: Variable



### 14.1.3 Creation of a “Function”

Figure 14.3: Function



To create a new function, click on the “New” button. Fill up the field Name then choose Function for the field “Symbol type”. Enter the body of the new variable. Enter also the parameters for the new function. On the top right, type a name for the new parameter then click on the “Add” button, just below this text field. Repeat this action until you have entered all the parameters. Click on “Add” to validate the creation of the new function (bottom left). You can modify by clicking on the list (top left) after selecting Variable in the Symbol defined. The selected variable appears with its value on the screen, change the fields then click “Modify”.

## 14.2 Declare in each model the parameters

### 14.2.1 Fault tree - Parameter for an event



Click on the right mouse button on an event, and then select “Parameter”. A frame appears on the screen. Select the distribution function you want for the selected event.

Figure 14.4: Distributions available for a fault tree model

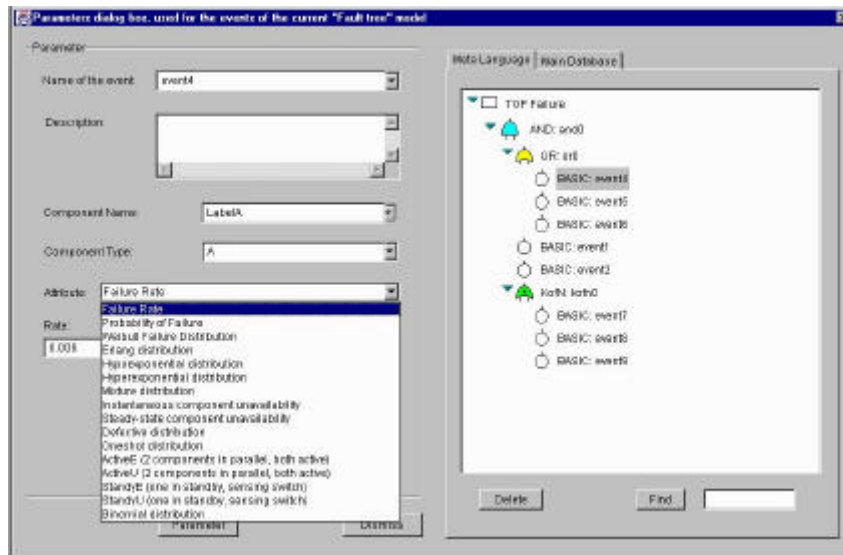
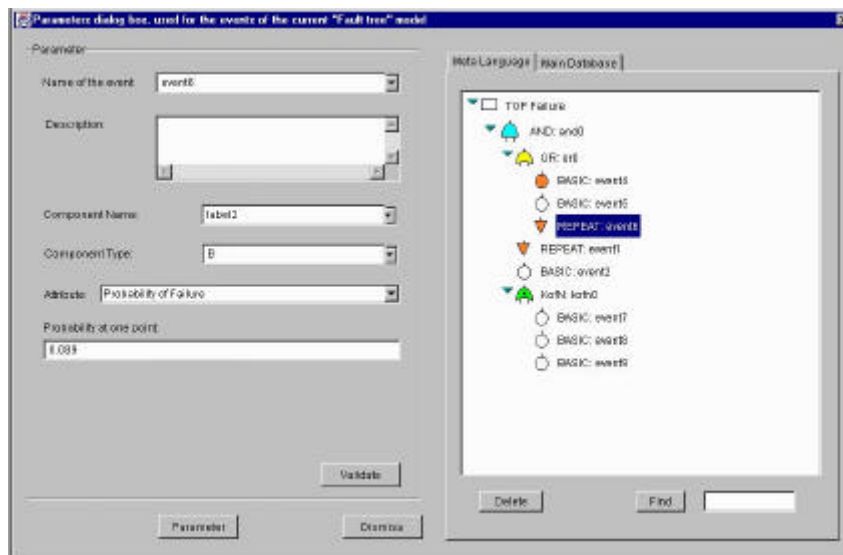


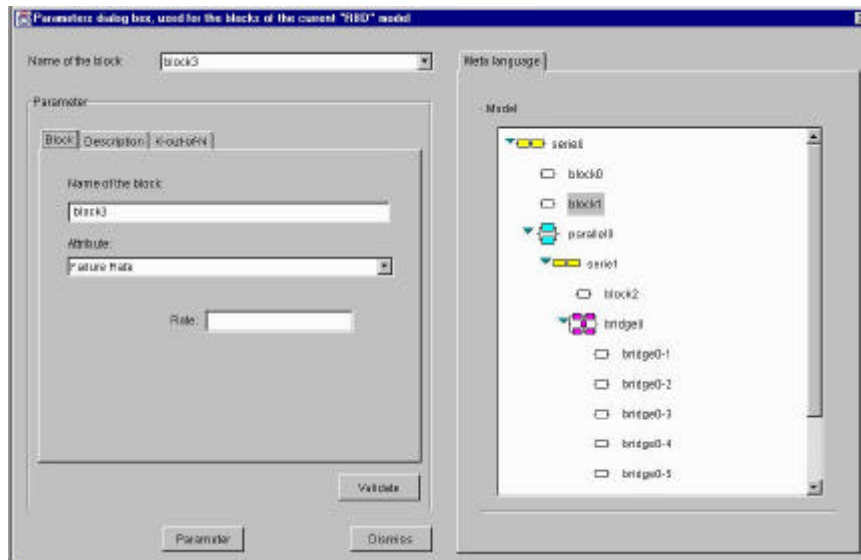
Figure 14.5: Metalanguage



### 14.2.2 Reliability block diagram - Parameter for a block

Click on the right mouse button on an event, and then select “Parameter”. A frame appears on the screen. Select the distribution function you want for the selected block.

Figure 14.6: Parameter's frame for RBD



### 14.2.3 Markov chain - Parameter for arc

For a arc: Click on an arc with the right mouse button, then click on “Rate” or click with the right mouse button on the panel, then on “Rate matrix”.

### 14.2.4 Generalized stochastic Petri net - Parameter for place/transition/arc

For a place: Click on a place with the right mouse button, then click on “Property”.

For a transition: Click on a transition with the right mouse button, then click on “Property”

For an arc: Click on an arc with the right mouse button, then click on “Properties”.

### 14.2.5 Product-form queuing network - Parameter for station/arc

For a station: Click on a station with the right mouse button, then click on “Properties”.

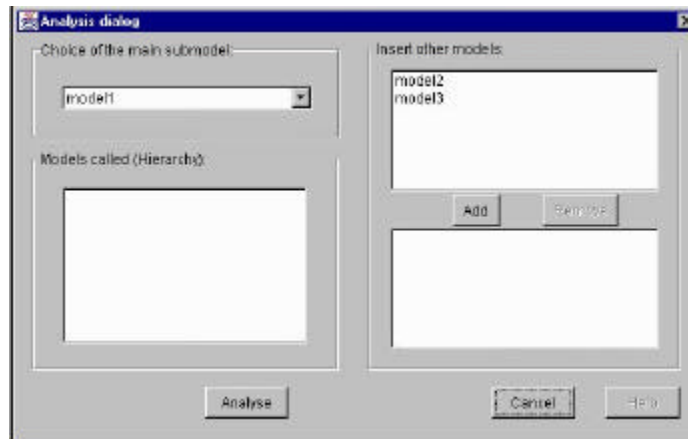
For an arc: Click on an arc with the right mouse button, then click on “Property”.

# Chapter 15

## Analysis of a model

### 15.1 Analysis of the model

Figure 15.1: Analysis frame



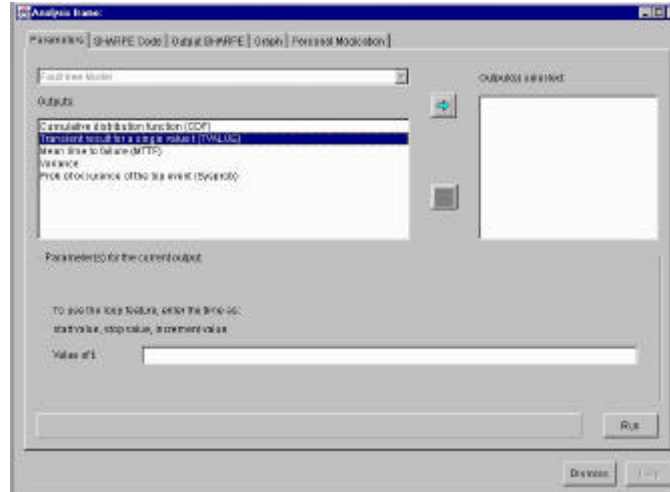
Analysis / Visualization of the results produced by SHARPE

Click on “Analysis Editor” then on “Analysis”. If the project has more than one model, the following window appears. You can choose which model is the main one to be analyzed for SHARPE. You can see in the left bottom part a list, which contained the models called from the one chosen to be the main model. On the right part, you can see on the top a list, which contains the other models of the current project. You can add some models in the code by selecting one model at a time in this list then clicking on the “Add” button. The reverse action, removing a model from the current code is done by clicking on the “Remove button”.

Click on “Analyze” to continue in the analysis. Click on “Cancel” to come back to the design. After clicking on “Analyze”, the following frame appears. A tabset panel, inside this frame, is composed of 5 parts.

- Parameters
- SHARPE code
- Output SHARPE
- Graph
- Personal Modification

Figure 15.2: Selection of the outputs



## 15.2 Parameter not defined in the main model, Outputs chosen

On the top, you can see a list contained all the output functions related with the type of the main model. Click on one of the outputs then click on the right blue arrow to add the selected output to be analyzed. Some outputs required some parameters like for the function tvalue, you need to enter a value for the time t. A message will appear on the message bar if you didn't enter all the values for the selected output. Also if the model is calling some variables not defined, a grid appear on the right bottom part, expecting the user to complete the values of each parameter for any output added to be analyzed. To give value for the parameters, click inside the grid in the column 2, the cell becomes black so you are ready to enter a value.

Then after you have finished to choose the outputs expected, click on "Run" to call SHARPE. At the end of the execution of SHARPE, the GUI shows the results.

## 15.3 SHARPE code from the current model

The SHARPE code for the current model can be viewed.

Click on "Analysis Editor" and then on "Analysis". A frame appears on the screen. Click on "SHARPE code" in the tabset object.

Figure 15.3: Selection of the outputs

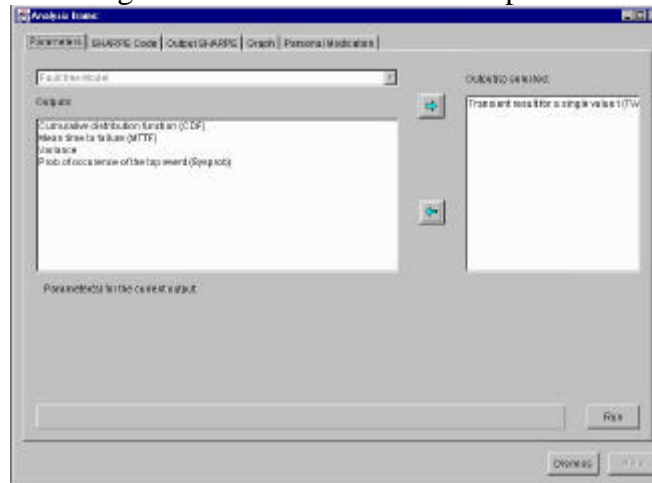
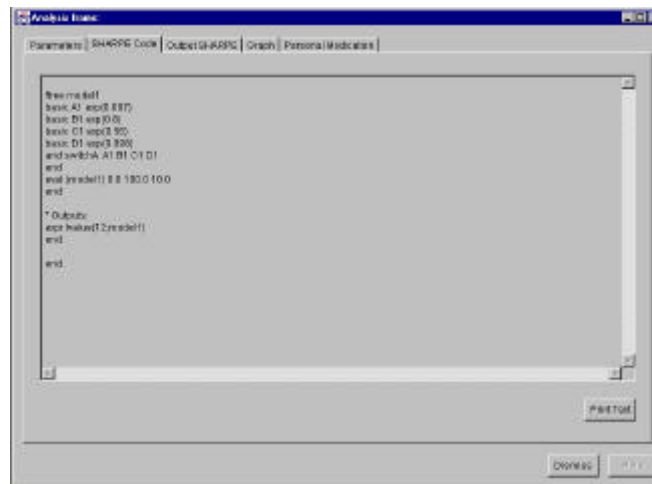


Figure 15.4: SHARPE code



## 15.4 SHARPE output from the current model

The SHARPE output for the current model can be viewed. Click on “Analysis Editor” and then on “Analysis”. A frame appears on the screen. Click on “output SHARPE” in the tabset object.

## 15.5 Graph

The interface has the ability to represent SHARPE output as a plot. When “Analysis Editor” then “Analysis” is clicked, a frame appears on the screen. Click on “Graph” in the tabset object. Fill up the following fields:

- Name of the graph

- Start value
- Stop value
- Increment value

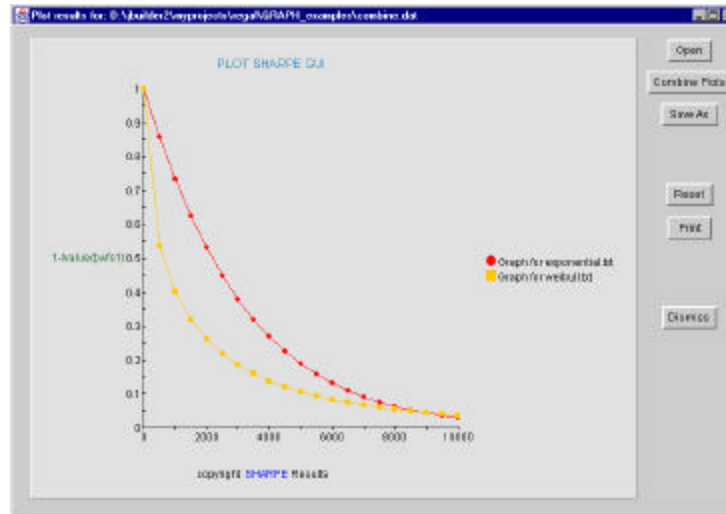
After completing all the fields, click on “Run”. The plot appears in a frame. If the main model contains parameters not defined, the button “Run” is disabled, click on the button “Parameter” to enter the value of the parameters, the button “Run” will be enabled. The plot can be saved in postscript format by clicking on “Print Plot”.

## 15.6 Personal modification of the SHARPE code

After the design phase, personal modifications to the code can be made in the “Personal Modification” tab. You can save the code in a text file by clicking on the button “Print Text”.

# Chapter 16

## Plotting the results of a model



### 16.1 Open a plot

Click on the button “Open”. The graph files have the extension .dat.

### 16.2 Combine a plot

You can combine many plots in one plot if any plots which will be opened are in the same interval  $[x1,x2]$  when the same number of points. Click on the button “Combine” to add another plot to the current one. The graph files have the extension .dat.

### 16.3 Print a plot

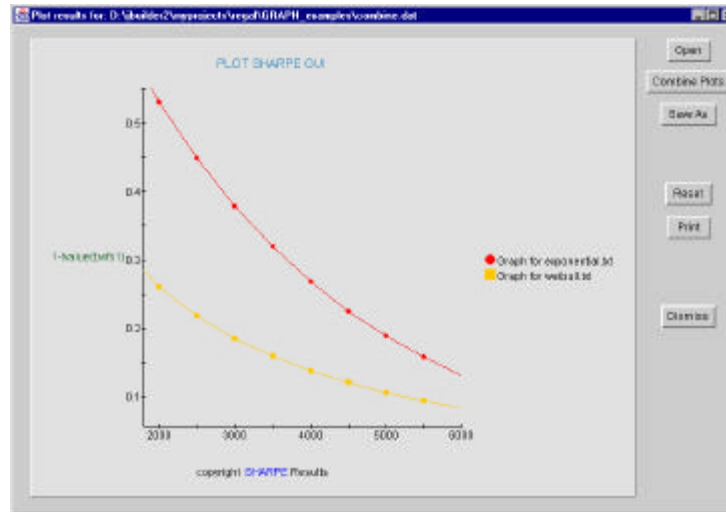
Click on the button “Print” to print the current plot.

### 16.4 Save a plot

Click on the button “Save As” to save the current plot.

## 16.4 Zoom the plot

To zoom on a specific area, click on the plot when the key shift pressed, then drag the mouse to create a rectangle which will define the area you want to zoom.





# Chapter 17

## Help features inside the GUI

### 17.1 Use of two statusBars:

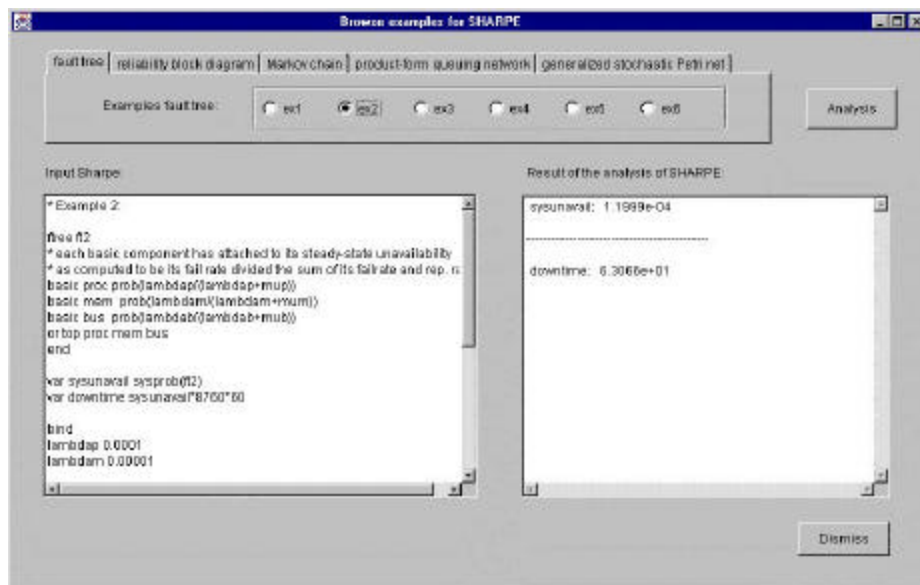
The interface has 2 status Bars located on the top of the main frame. During the session, some messages will appear to help the user during his design.

### 17.2 Definition of the button Bars

For each type of model, the buttonBar look is different. If the user pass over a button, the description of the button will appear in the right status Bar.

### 17.3 Browse Frame

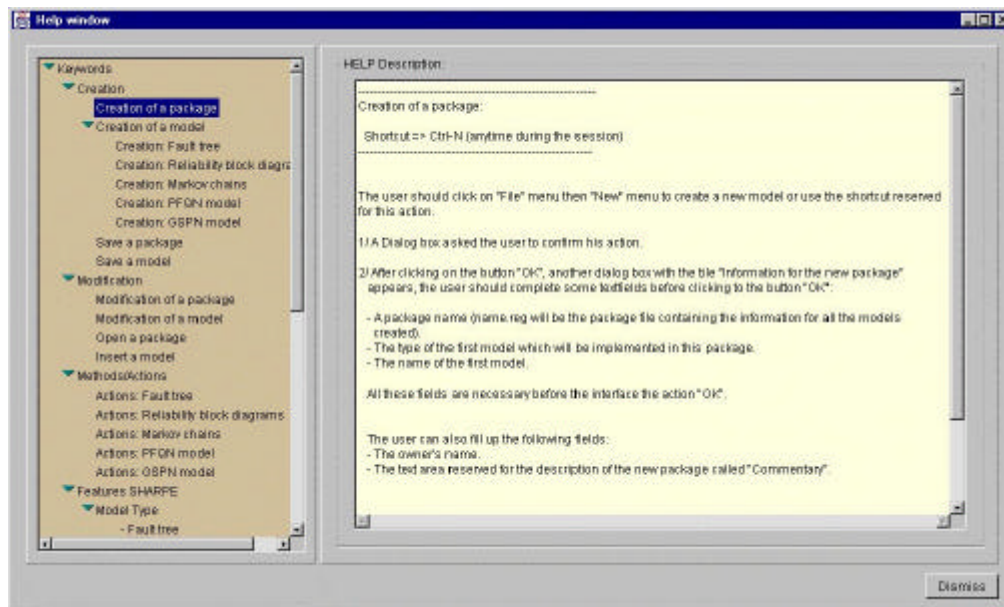
Figure 17.1: Browse frame



The Browse button produces the frame shown in Figure 18. This frame gives the details of all the names, attributes, and values which are used in the model. It also displays how many events use each as its property. Another important aspect of the browse frame is that it will alert the user as to how many events have yet to have their property defined. This is important because all events need to be defined before an evaluation can be performed.

## 17.4 Help Frame

Figure 17.2: Help frame



The purpose of the Help page is to provide online help for the user during the use of the interface. The Help page is in a hypertext format such that when the user selects certain keywords in the text, the Help page is changed to reflect the information of the selected keyword.