

Title
Integration Rational Rose ↔ TakeFive SNIFF+

Integration of Rational Rose with SNIFF+

Rose - side

a) The following actions are defined for the integration with SNIFF+. Each of them can be executed with the menu **Tools** -> **SNIFF+**.

- **Edit Class**
Loads the class selected in a class diagram into the Source Editor (SE) from SNIFF+.
corresponding script: **EditClass.ebx**
- **Show Entire Hierarchy**
Loads the class selected in a class diagram into the Hierarchy Browser (HB) of SNIFF+, automatically selected and displayed with all other classes.
corresponding script: **FullHier.ebx**
- **Show Relative Hierarchy**
Loads the class selected in a class diagram into the Hierarchy Browser (HB) of SNIFF+, automatically selected and displayed with all directly associated classes.
corresponding script: **RelHier.ebx**
- **Show Class Browser**
Loads the class selected in a class diagram into the Class Browser (CB) of SNIFF+.
corresponding script: **BrowseClass.ebx**

b) The execution of a predefined action is handled by embedding into Rose menus. The following structure describes the menus:

- A submenu **SNIFF+** is located in the **Tools** menu.

The definition of the menu structure can be found in the file **rose.mnu** which is located in the installation directory of Rose (i.e. where rose.exe is located). The following changes have to be made:

```
Menu Tools
{
  # optional entries
  Separator
  #...
  menu "SNIFF+"
  {
    option "&Edit Class ^""%selected_items"^^"
    {
      enable %selected_items:unary
      RoseScript $SENS_TOOLS\
```

Title
Integration Rational Rose ↔ TakeFive SNiFF+

```
option "Show Symbol in Cross Referencer" ^"%selected_items"^^"  
{  
    enable %selected_items:unary  
    RoseScript $SENS_TOOLS\<<corresponding script>  
}  
}
```

- The environment variable %SENS_TOOLS% should contain the path, where the additional programs from SENS are located.
- c) The communication from Rose to SNiFF+ is done with Rose BasicScripts. These scripts define the selections in a diagram and start actions by calling the SNiFF interface program **sniffaccess** with the corresponding parameters.
- d) Installation:
- The environment variable %SENS_TOOLS% must be initialized with the corresponding path:
WinNT: Control panel → System → Environment → <variable and value>
Rose: File → Edit Path Map... → <symbol and actual path>
 - Copy corresponding scripts to %SENS_TOOLS%
 - Add menu structure in <rose-inst-Dir>\rose.mnu as described in section b)

SNiFF+ - side

- a) The following actions are defined for the integration of Rose. Those actions can be executed by using the **RationalRose** menu from within the Project Editor (PE), Source Editor (SE) or the context menu (right mouse click) in the Project Editor.

Project Editor:

- **Start Rose**
The application Rose will be started.
corresponding script: -
- **Open Model**
A selected model – file (.mdl) will be loaded into Rose. The file extension will be checked if it is equal .mdl!
corresponding script: -
- **Show Component Diagram**
A component diagram containing the selected file will be called in Rose. If more diagrams are available in Rose, an additional menu provides the possibility to select one.
corresponding script: **FileInDgrm.ebx**
- **Quit Rose**
The application rose will be closed.
corresponding script: -

Source Editor:

- **Start Rose**
The application Rose will be started.
corresponding script: -
- **Show Class in Diagram**
A class diagram containing the selected class will be launched. If more diagrams are

Title
Integration Rational Rose ↔ TakeFive SNiFF+

available in Rose, the desired one can be selected through an additional menu.

corresponding script: **ClassInDgrm.ebx**

- A class diagram containing the selected method will be launched. corresponding A class diagram containing the selected attribute will be launched. corresponding Quit Rose The application Rose will be closed.

corresponding script: -

Context Menu:

- **Open Model**

A selected model – file (.mdl) will be loaded into Rose. The file extension will be checked if it is equal .mdl!

corresponding script: -

- **Show Component Diagram**

A component diagram containing the selected file will be called in Rose. If more diagrams are available in Rose, an additional menu provides the possibility to select one.

corresponding script: **FileInDgrm.ebx**

- b) The execution of a predefined action can be done by inserting of an additional menu item. The **Context Menu** can be adapted with the Site Level settings or the project attributes. The menu bar of the Project Editor and / or the Source Editor can be modified by editing the file **SiteMenus.sniff** in the directory %SNIFF_DIR%\config. The following modifications have to be made:

```
#####
# SOURCE EDITOR #
#####

^Editor
... # additional entries
#
>Rational&Rose
shell "Start Rose ..." "$SNIFF_DIR\bin\toRoseIF -s"
shell "Show Class Diagram..." "$SNIFF_DIR\bin\toRoseIF -c %S "
shell "Show Class with method..." "$SNIFF_DIR\bin\toRoseIF -m %S "
shell "Show Class with attribute..." "$SNIFF_DIR\bin\toRoseIF -a %S "

# optional additional options
-
YesNoDialog "Quit Rose..." "$SNIFF_DIR\bin\toRoseIF -x " "OK to quit Rational Rose ?"

#####
# PROJECT EDITOR #
#####

^ProjectEditor
... # additional entries
#
>Rational&Rose
shell "Start Rose ..." "$SNIFF_DIR\bin\toRoseIF -s"
shell "Open Model \"%S\" " "$SNIFF_DIR\bin\toRoseIF -o %F"
# optional additional options
-
YesNoDialog "Quit Rose..." "$SNIFF_DIR\bin\toRoseIF -x " "OK to quit Rational Rose ?"
```

The **Context Menu** settings are defined in the preferences. Individual parameters for titles, actions, icons etc. in the **Context Menu** are defined for each file type.

Title
Integration Rational Rose ↔ TakeFive SNiFF+

The following table shows the correlation between Rational Rose file extensions and file types.

File Extension	Icon File	Preference File	Description
.mdl	IconMdl.gif	RoseModel.sniff	Describes the whole model and is always controlled unit
.cat	IconCat.gif	RoseCategory.sniff	controlled unit type "logical package"
.sub	IconSub.gif	RoseSubsystem.sniff	controlled unit type "component package"
.prc	IconPrc.gif	RoseDeployment.sniff	"deployment diagram" as controlled unit
.prp	IconPrp.gif	RoseProperties.sniff	"sets of model properties" as controlled unit

c) The communication from SNiFF+ to Rose consists of two steps:

In the first step, the interface application **toRoseIF.exe** which is located in the directory **%SNIFF_DIR%\bin** is executed with some additional parameters. It establishes an OLE – connection to Rose, passes the parameters to the registry and executes the first script **IFDispatch.ebx** in the directory **\$SENS_TOOLS**.

In the second step, this script reads the parameters from the registry. The parameters define which action is to be done and which of the Rose BasicScripts are to be executed.

The following table describes the parameter syntax for the application **toRoseIF.exe**:

ToRoseIF.exe -a <attribute-name>	Shows the class in a class diagram containing the attribute <attribute-name>
ToRoseIF.exe -c <class-name>	Shows the class diagram containing the class <class-name>.
ToRoseIF.exe -f <source-file>	Shows the component diagram containing the file <source-file>.
ToRoseIF.exe -m <method-name>	Shows the class containing the method <method-name> in a class diagram.
ToRoseIF.exe -o <model-name>	Starts Rose and loads the model <model-name>
ToRoseIF.exe -s	Starts Rose
ToRoseIF.exe -x	Closes Rose

d) Installation:

- Copy the interface program **toRoseIF.exe** into the directory **%SNIFF_DIR%\bin**.
- Modify the file **%SNIFF_DIR%\config\SiteMenues.sniff** as described in section b).
- Initialize the environment variable **\$SENS_TOOLS** with the corresponding path:
WinNT: Control Panel → System → Environment → <variable and value>
Rose: File → Edit Path Map... → <symbol and actual path>
- Create a subdirectory **Icons** in the directory **%SENS_TOOLS%**.
- Copy the applications **IFDispatch.ebx** and the corresponding scripts into the directory **%SENS_TOOLS%**.
- Copy the icon – files (*.gif as described in section b)) into the directory **%SENS_TOOLS%\Icons**.
- Copy the preferences – files (*.sniff as described in section b)) into the directory **\$SNIFF_DIR\Preferences\FileTypes**.