

SNiFF+
Release 3.x
for Unix and Windows

Integration of Rational Rose and SNiFF+ for Windows



TakeFive Software GmbH
5020 Salzburg, Austria
eMail: info@takefive.co.at

TakeFive Software, Inc
Cupertino, CA
E-Mail: info@takefive.com

Integration of Rational Rose and SNIFF+ for Windows

Overview

Rational Rose provides a graphical software-engineering tool which supports the most popular OO analysis and design methods.

SNIFF+ is the unique multi-language multi-platform Source Code Engineering tool by TakeFive Software. It provides an integrated environment for all source code related tasks from code comprehension and browsing to team development and project management even with large software projects. The flexible, open architecture of SNIFF+ makes integrations with most popular software engineering tools possible.

This integration provides a bi-directional navigation between SNIFF+ and Rose/C++ for both source code that has been generated from a Rose Design and Rose models that have been created from source code by the Rose Analyzer.

Compatibility

This Integration has been developed with Rational Rose 98i and SNIFF+ 3.2. It was tested under windows NT4 SP6.

How it works

The communication from Rose to SNIFF+ is done through an ATL COM controller. This is done under a dll file (`rosetosniff.dll`). Rose calls methods by passing a dispatch pointer. This pointer points to an interface, which can collect information from Rational Rose. These methods call the `sniffaccess` program, which can drive SNIFF+.

The communication from SNIFF+ to Rose consists of two steps: In the first step, the program `snifftorose` is called from `sniff` with the specified parameter. In the second, this program tries to connect rational rose and calls the specified tool.

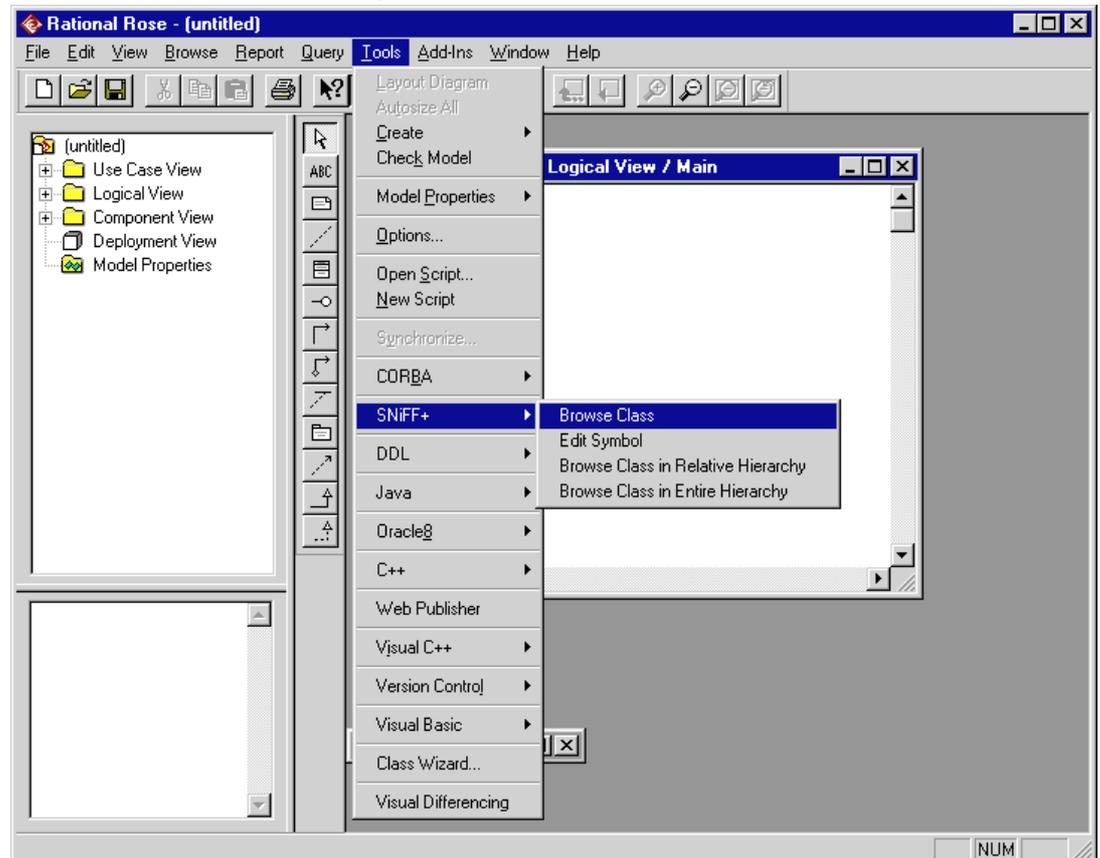
In the following documentation you can find a list of the actions implemented and how to install the integration.

Installation

- Copy the integration files to a place where all Rose and SNIFF+ users can access it.
- The environment variable `%ROSEHOME%` **must** be set to the root directory where Rose is installed (`<rational_dir>/Rose 98i`). This is only needed for starting the Analyzer.
- Execute the `install.exe` program. This installation program will install all the components and update the registry base.

Navigation from Rose to SNIFF+

An additional menu **Tools > SNIFF+** is added to Rational Rose for the integration with SNIFF+. Therein, the following actions are defined:



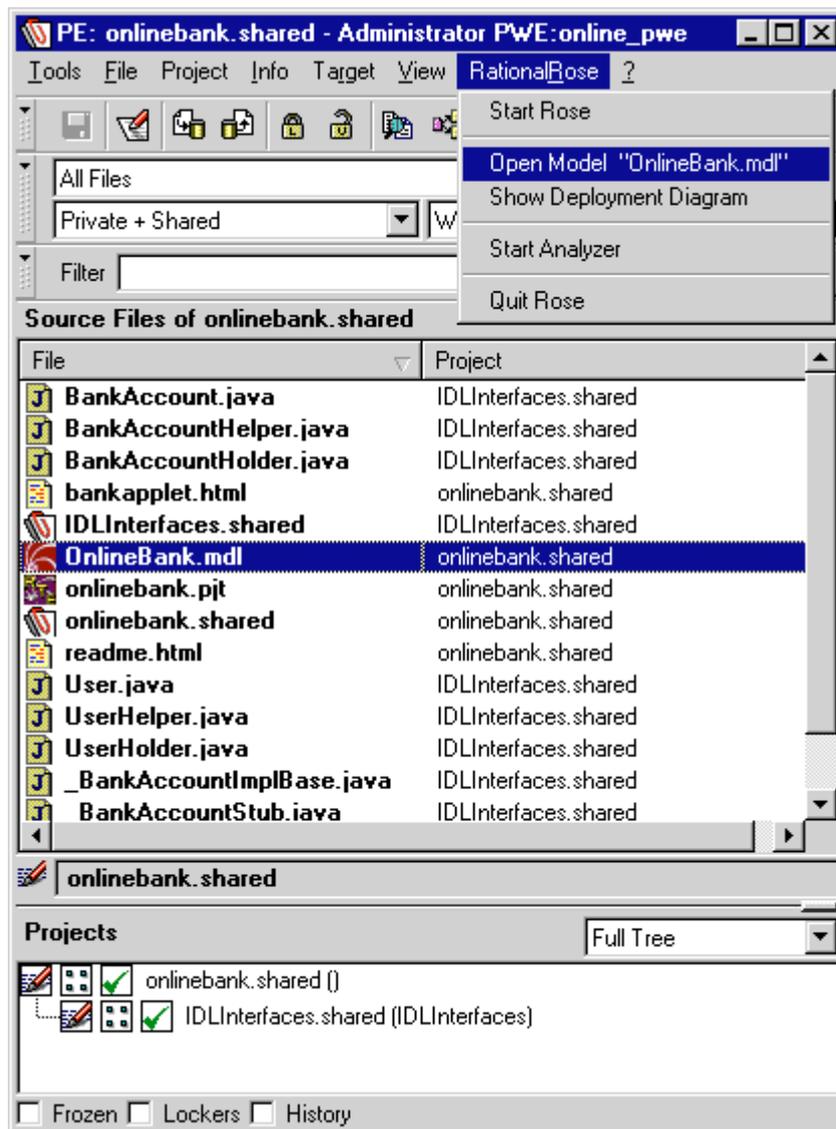
- **Browse Class**
Loads the class selected in a class diagram into the Class Browser (CB) of SNIFF+.
- **Edit Symbol**
Loads the class selected in a class diagram into the Source Editor (SE) of SNIFF+.
- **Browse Class in Relative Hierarchy**
Loads the class selected in a class diagram into the Hierarchy Browser (HB) of SNIFF+, automatically selected and displayed with all directly associated classes.
- **Browse Class in 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.

Navigation from SNIFF+ to Rose

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

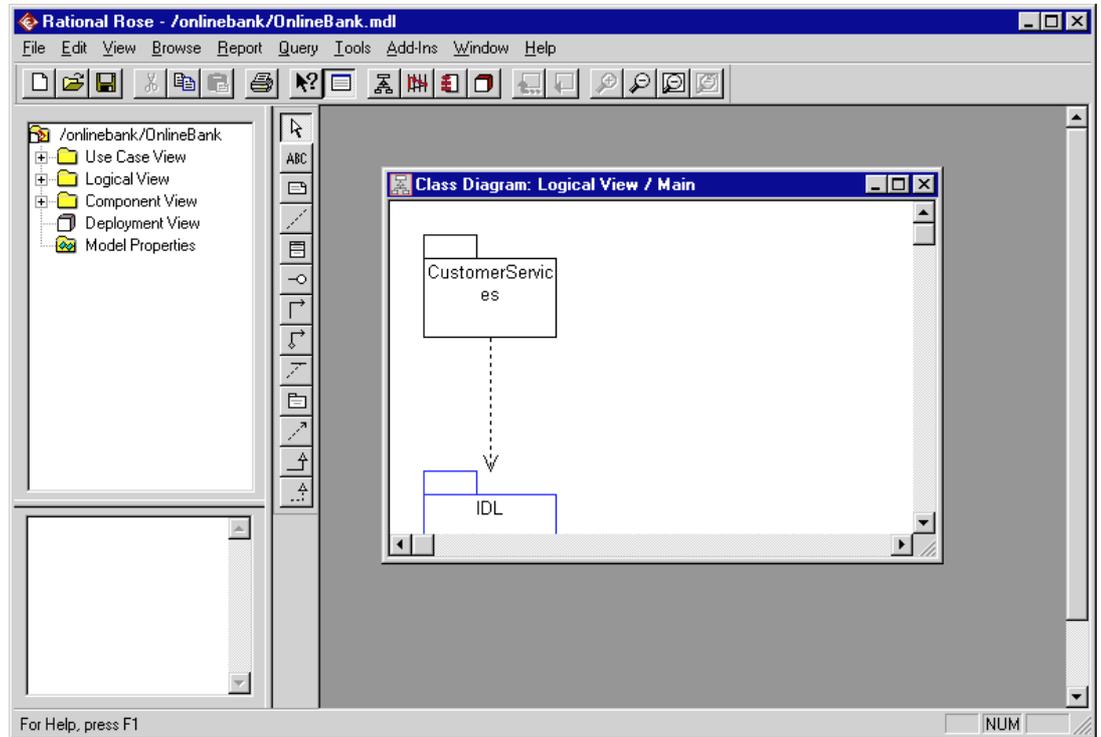
This screenshot shows the Project Editor with the new menu **RationalRose**.



- Start Rose
 - Rational Rose will be started.

- **Open Model "Model Name"**

A selected model – **Model Name.md1** will be loaded into Rose. The file extension will be checked if it is equal to **.md1**! If Rose hasn't been loaded yet, it will be started and the selected model will be shown in the class diagramm.



- **Show deployment Diagram**

A deployment diagram from the current rose model will be set to visible in rose.

- **Start Analyzer**

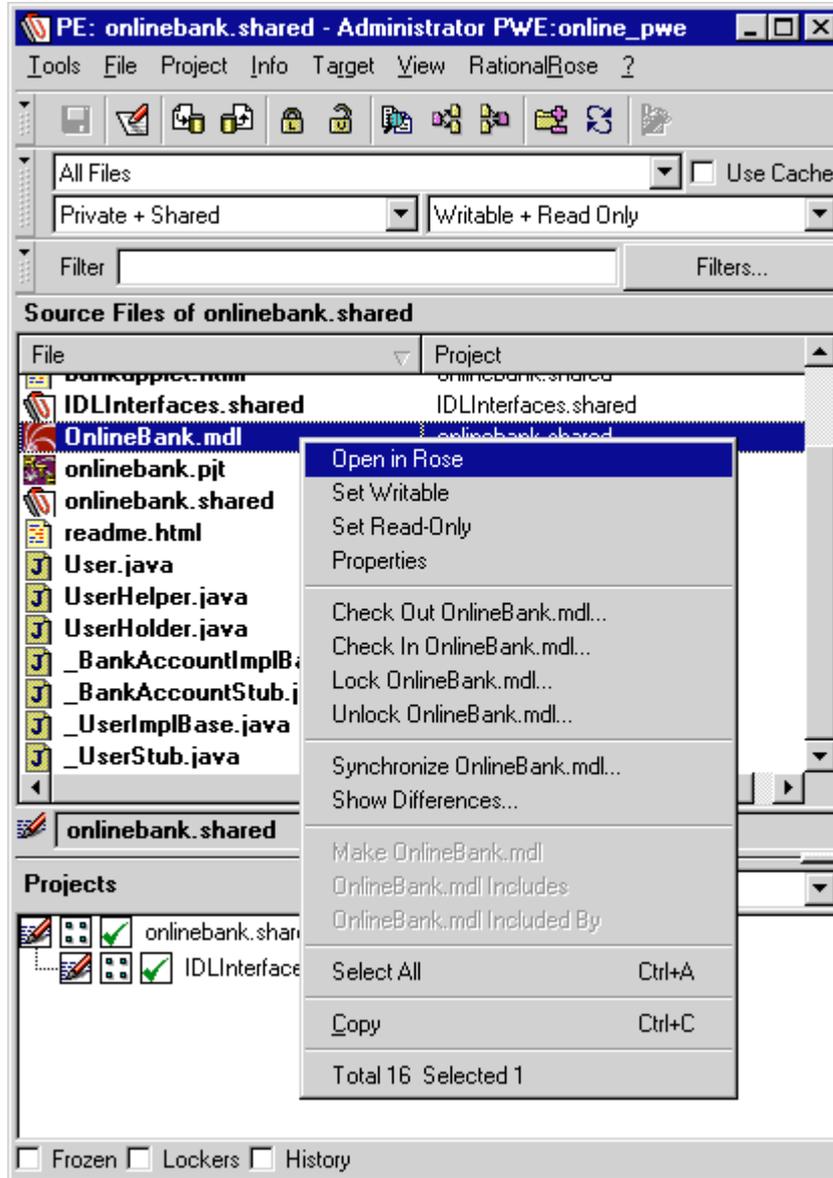
The Rational Rose C++ Analyzer starts.

- **Quit Rose**

Rational Rose will be closed.

Project Editor Context Menu

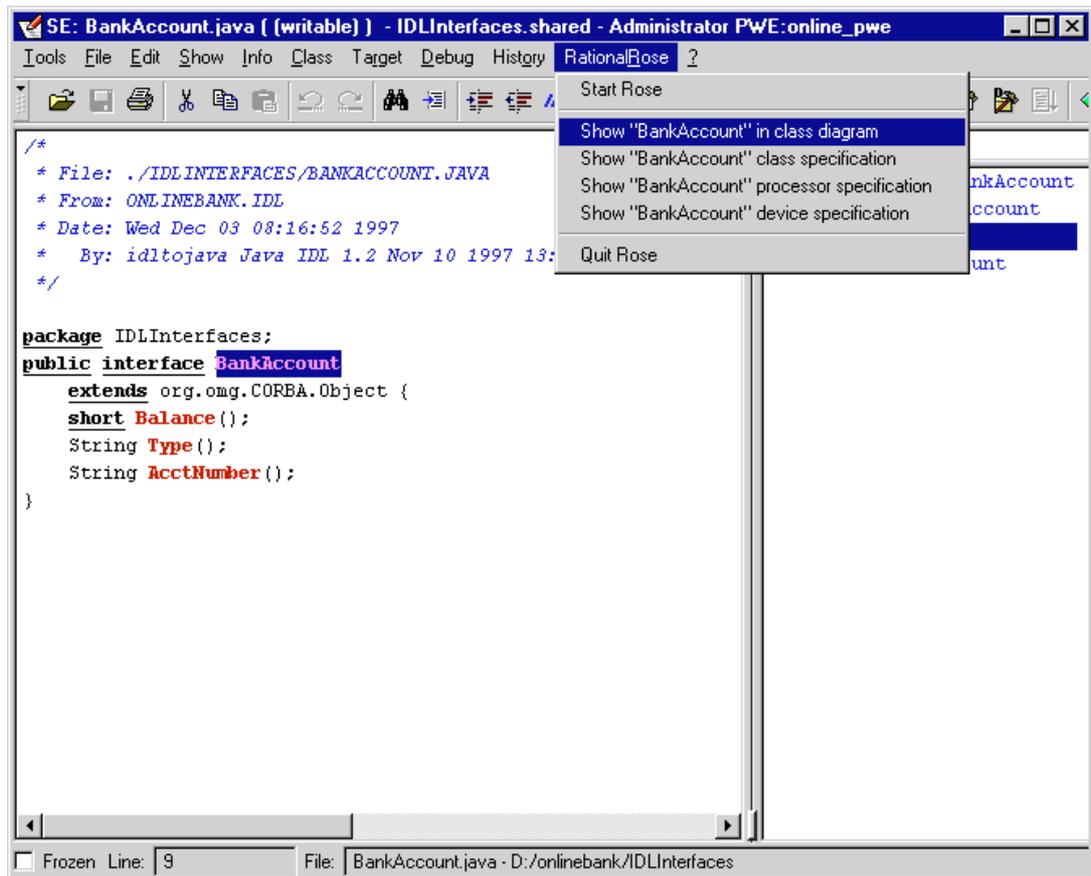
When you select a model- or an analyzer-file you can activate the context menu:



- **Open in Rose**
Tied to the Rose model (.mdl) file type. A selected model – file (.mdl) will be loaded into Rose.
- **Open in Analyzer**

Tied to the Analyzer Project (.pjf) file type. A selected .pjf file will be opened with the Rose Analyzer.

Source Editor



In order to browse a symbol from the Source Editor in Rose, the whole symbol to be browsed needs to be selected first. This can be accomplished by double-clicking on the symbol to browse, or by using the speedbar on the right-hand side of the editor to select a symbol.

- **Start Rose**

Rose 98i will be started.

- **Show *symbol* in class diagram**

A class diagram containing the selected class will be launched. If more diagrams are available in Rose, the desired one can be selected through an additional menu.

- **Show *symbol* class specification**

The Rose specification window for the corresponding class will be shown. From the specification window, the "Browse" button can be used to find the corresponding class in the browser or in a diagram.

- **Show *symbol* processor specification**
The Rose specification window for the corresponding processor will be shown.
- **Show *symbol* device specification**
The Rose specification window for the corresponding device will be shown.
- **Quit Rose**
Rational Rose will be closed.

Technical Details

Menu Definition in Rose

The execution of a predefined action is handled by embedding into Rose menus. The definition of the menu structure can be found in the file `rosetosniff.mnu` which is located in the `SNIFF_DIR` installation directory:

```
Menu Tools
{
    Menu SNIFF+
    {
        Separator
        option "Browse Class"
        {
            InterfaceEvent Rosetosniff callclassbrowser
        }
        option "Edit Symbol"
        {
            InterfaceEvent Rosetosniff calleditsymbol
        }
        option "Browse Class in relative Hierarchy"
        {
            InterfaceEvent Rosetosniff callrelativehierarchy
        }

        option "Browse Class in entire Hierarchy"
        {
            InterfaceEvent Rosetosniff callentirehierarchy
        }
    }
}
```

Menu Definitions in SNIFF+

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:

```
#####
#Rose integration menu definitions
#####
#   Project editor           #
#####
^ProjectEditor
>Rational&Rose
shell "Start Rose" "$SNIFF_DIR\bin\snifftorose"
-
shell "Open Model  \"%S\"" "$SNIFF_DIR\bin\snifftorose -open_model
 \"%f\""
shell "Show Deployment Diagram" "$SNIFF_DIR\bin\snifftorose -
open_deployment_diagram"
-
shell "Start Analyzer" "\"$ROSEHOME/c++/analyzer.exe\""
-
shell "Quit Rose" "$SNIFF_DIR\bin\snifftorose -exit_rose
#####
#   SOURCE EDITOR           #
#####
^Editor           # uncomment the beginning of this line to activate the
menu
>Rational&Rose
shell "Start Rose" "$SNIFF_DIR\bin\snifftorose"
-
shell "Show \"%k\" in class diagram" "$SNIFF_DIR\bin\snifftorose -
view_class_diagram %S"
shell "Show \"%k\" class specification" "$SNIFF_DIR\bin\snifftorose
-open_class_specification %S"
shell "Show \"%S\" processor specification"
"$SNIFF_DIR\bin\snifftorose -open_processor_specification %S "
shell "Show \"%S\" device specification"
"$SNIFF_DIR\bin\snifftorose -open_device_specification %S "
-
shell "Quit Rose" "$SNIFF_DIR\bin\snifftorose -exit_rose
```

New SNIFF+ File Types and associated actions

The **Context Menu** settings are defined in the File Type Preferences. Individual parameters for titles, actions, icons etc. in are defined for each file type. For each file type, a preferences file is supplied.

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"
.pjt	IconPjt.gif	RoseAnalyzer.sniff	Project Description File for Rose C++ Analyzer
.red	IconMdl.gif	RoseRedcode.sniff	Reengineered Code from Rose C++ Analyzer

Known Problems

- When navigating from Rose to SNIFF+, the selected SNIFF+ tool does not always pop to the front.
- File types for Rose Property Units (.pp), Subsystems (.sub) and controlled units (.prc) have not yet been defined.
- Names in Rose that contain more than one word (i.e. that have spaces inside) cannot be associated with their counterparts in the code (which are usually generated by Rose to have underscores instead of the spaces)
- The Rose Menu Entries "Browse Header" and "Browse Body" do not yet use the SNIFF+ Editor.
- The SNIFF Editor should be capable of hiding rose-generated active comments.