

SmallCOM/X

Specification

SmallCOM/X is a Win32-Tool, which generates VisualWorks-Smalltalk classes for using COM-, OLE-, ActiveX-, OCX- and .NET-, .NET-Forms components. In the subsequent description the term *Visual* component is used for an ActiveX-, OCX- and .NET-Forms-component. The term *NonVisual* component is used for a COM-, OLE- and .NET-component.

- **Why is SmallCOM/X developed**

In the Microsoft world many useful *Visual* and *NonVisual* components are existing. Using these components in your VisualWorks application, may save much time.

- **What does SmallCOM/X generate**

SmallCOM/X generates classes and code to handle *NonVisual* and *Visual* components. The *Visual* components are useable as normal widgets and can be chosen in the UIPainter through the generated artefacts:

- Spec-, View- und Controller classes.
- Integration into the UIPainter as a widget.
- Extensions to the UIBuilder.
- Classes for CoClasses, interfaces and types.
- Protocol for using the visual components as a COM component.

With these generated artefacts (classes and methods) the *Visual* component can be used like a normal widget too.

NonVisual components created artefacts:

- Classes for CoClasses, interfaces and types.
- Protocol for using the visual components as a COM component.

- **Requirements for VisualWorks**

You need a VisualWorks version supporting namespaces (5i.4 or higher) and the installed ComConnect parcel. The image must be running with a SmallTalkCommander (see *SmallCOM/X* documentation).

- **Requirements for the platform**

Because the codegenerator part of *SmallCOM/X* is a Win32-Tool, you need one of the Windows platform (Windows 95/98/NT/2000/2003/ME or XP). For .NET components you need a .NET enabled platform.

- **Requirements for the component**

The component must be properly installed on the system. .NET components must be made COM enabled. This is done with some options in the IDE and a few additional lines of code (see *SmallCOM/X* documentation).

- **How do you use SmallCOM/X**

First you select a component or component's typelibrary. This library is the most times registered in the Windows Registry, but could also be a .tlb-, .olb-, .dll or .exe-file. After loading the library, all founded types are shown (Enums, Interfaces and CoClasses). Additional the needed Spec-, View- und Controller-classes are shown. For each type a tab-view with the class and code to be generated is displayed. This code can be modified manually before filing in. Selected parts or all the parts can be filed in into a VisualWorks image.

After filing in, you can place a *Visual* component's widget with the new widget button in the UIPainter into a canvas and use it like the other widgets. If classes for a *NonVisual* component are created, you can communicate with the component by creating an instance of the generated class and send it some of the generated messages. Sending a message to an *Visual* component is also possible.

JOOPS Informationstechnik GmbH

Phone: ++49 (0)89-600 692-0

mailto: info@joops.com

Web: www.joops.com