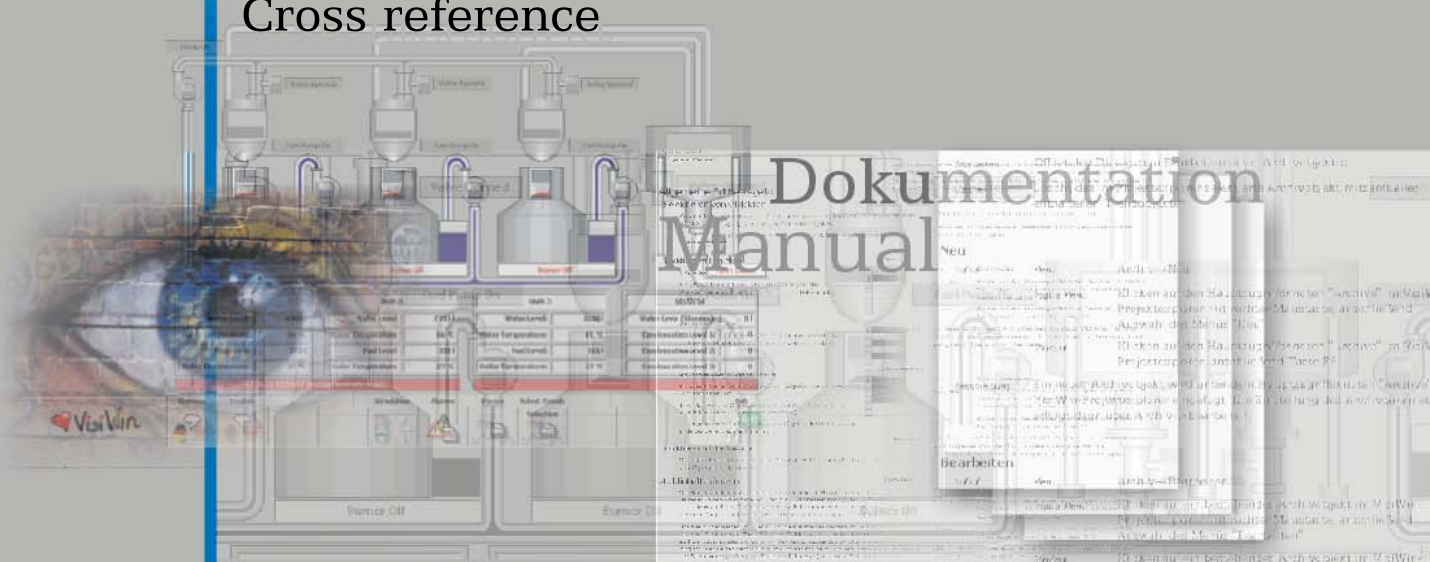


VisiWinNET 2005

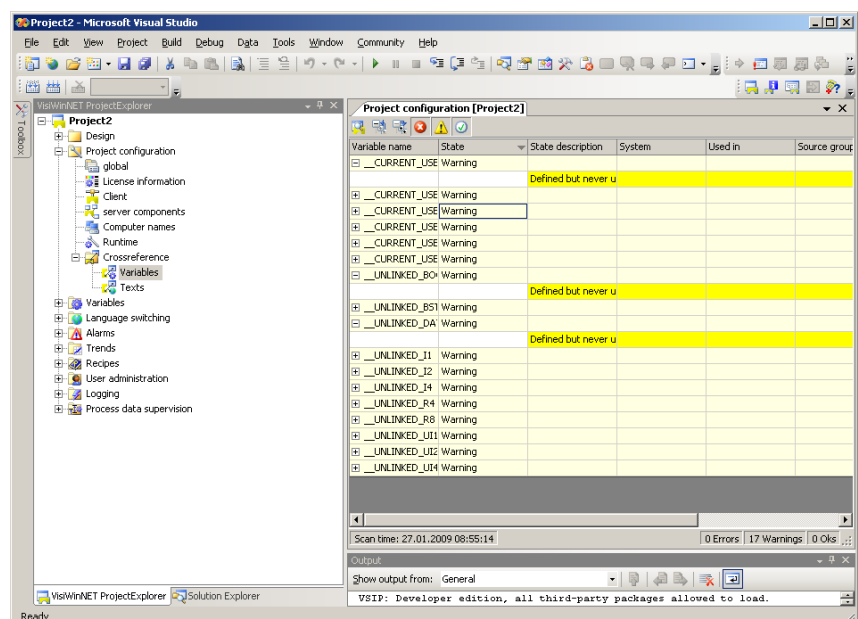
Cross reference



- **VisiWin**
- **VisiWinNET 2005**
- Common
- Class Library
- Systems
- **Tools**
- Technical Informations
- **Inosoft OPCServer**
- Basics and helping tools
- Protocols

VisiWinNET 2005

CROSS reference



Cross reference integrated in the development environment







The contents of this manual must not otherwise be used without explicit written consent from INOSOFT GmbH.

We have checked the contents of this manual for compliance with the described software. Discrepancies can, however, not be ruled out. For this reason we cannot guarantee full compliance. The contents of the manual are subject to regular checking for necessary updates/amendments. Such amendments will be made in the subsequent edition.

Suggestions for improvement are welcome.

Legend

In order to point out particular paragraphs the following symbols are used in the INOSOFT documentations:

	Attention	Passages with this sign should be read – and observed – with particular attention.
	Hint	Important paragraph “additional information”
	Tip	Many roads lead to Rome; here a shortcut is to be found.
	In work	Functions that are in preparation or already implemented but not yet prepared for documentation.
	Example execute	Instructions to be carried out in an example
	Observe result	Results to be observed with carrying out the exemplary instructions

© / ™ / ®

Windows®, Windows NT®, Windows 2000®, Windows XP® are registered trademarks of the Microsoft company.

Further product names marked ® are trademarks of the appropriate manufacturers.

INOSOFT GmbH created on

VisiWinNET Version: from 6.04.000

created on 07.06.2010

Contents

1 Preamble	1
2 Cross Reference	2
2.1 Variable Display	4
2.2 Text Display	5
2.3 Tool Bar	5
2.4 Status Descriptions.....	6

1 Preamble

About this manual

This manual contains information on the VisiWinNET cross reference. The cross reference analyzes the variables and texts used in the project. On one hand it shows the developer where which definition is to be used. On the other hand it reveals potential error sources that can typically creep in during a project development.

Questions and Problems

For technical questions and problems please contact your responsible INOSOFT agent or the INOSOFT GmbH Support under +49 (5221) 16 66 02 or email: Support@INOSOFT.com

Frequent questions and problems are dealt with on our homepage under www.inosoft.com

There you will also find a support area for direct contact with our Main Office.

2 Cross Reference

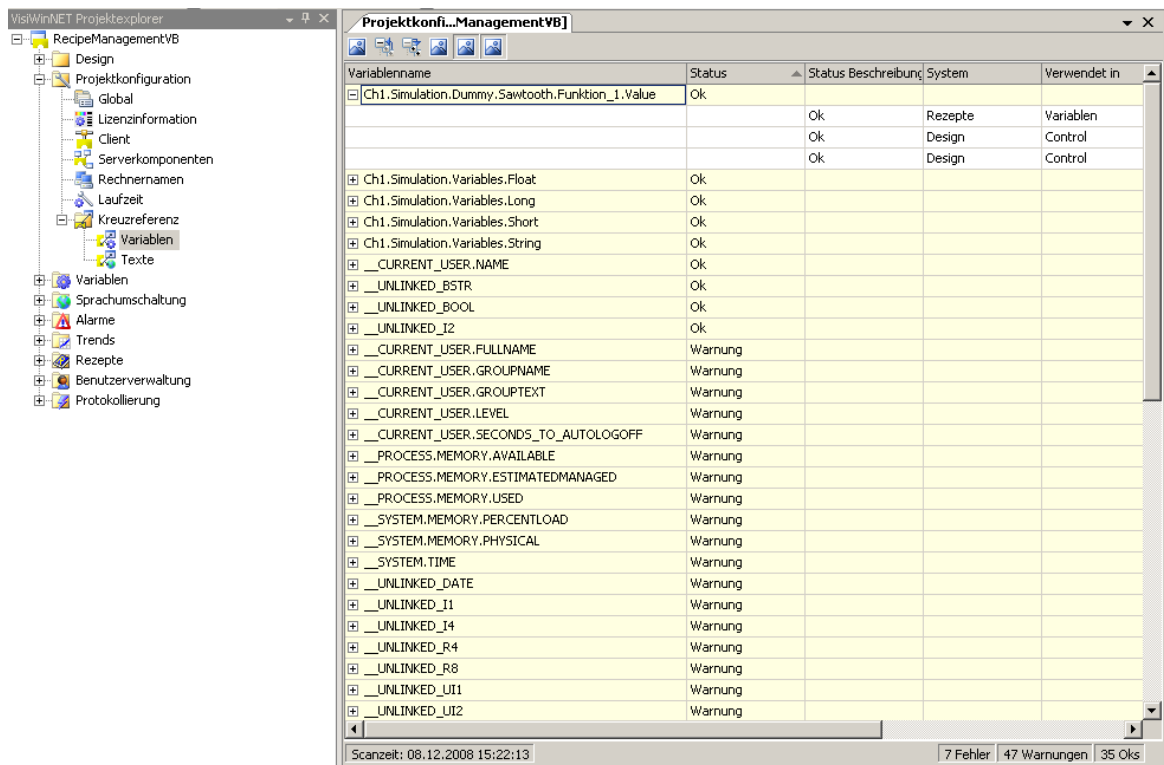
The VisiWinNET cross reference as an auxiliary tool provides a summary of the process variables and texts used in the application. As an example unused or not existing definitions can be listed with the cross reference.

As a diagnosis tool the cross reference also lists potential error sources and redundant definitions. In addition it is shown where which definition is used.

Start of the cross reference

The display of the cross reference is to be opened in the Project Explorer through the "Project configuration→Cross reference" node.

For the cross reference to display data first use the "Scan" button in the tool bar area of the window.



Changing the display

To change between the "Variables" and "Texts" lists in the cross reference the appropriate sub-node in the Project Explorer can be clicked on.

Searching for errors

There are three switches in the tool bar that each filter the display for a status:

- Error
- Warning
- OK

A click on one of the switches shows or hides the datasets with the appropriate status. This means that if only the errors or warnings are to be shown the OK switch must be clicked on.

When one or more errors in the application have been rectified the Scan process must be re-started to update the cross reference data. Prior to this the changes forms must be saved.

Searching for a variable / text

Even if there is no search function within the cross reference the lists can be sorted alphabetically. If for example the name of a process variable is to be searched for the appropriate column head must be clicked on.

Checking the use of a definition

More details are shown in the display if the developer clicks on the + symbol in the first column. In the secondary lines of the table the positions of the project where the variable or text was used are shown in detail.

2.1 Variable Display

The variable display compares the use of variables in the application with the definition from the project databank. Here it is shown where which variable is addressed in the application and where if applicable a use error was detected.

Column	Description
Variable name	Full designator of the variable (as used in the control elements)
Status	Allocation to an error category, either "OK", "Warning" or "Error"
Status description	Detailed information on the status (see chap. Status description)
System	Name of the system in which the variable is used. If "Design" is specified as system the variable is used in the design area (as a property value or by the program)
Used in	Specification of the definition in which the variable is used as a parameter or indication what use the variable has in the application

The following entries are possible for "Used in":

Name of a definition type	"Alarm" for example indicates that the variable is used as a parameter in an alarm definition.
Control	Indicates that the variable is used as a property value in a control element.
VWGet	Indicates that the item is used in a 'VWGet' access (direct reading of a variable value in the source code without link to a control element)
VWSet	Indicates that the item is used in a 'VWSet' access (direct writing of a variable value in the source code without link to a control element)
Source group/Source/Source detail	Further details on the access

2.2 Text Display

The text display shows the use of text by the property values and source code accesses of the application.

Column	Description
Text name	Specified the full designator of the text
Text type	Specifies the type of the text (specified through the uppermost text group)
No text	A check box activated here points to an incomplete translation. In at least one of the three language fields there is no text specified.
Status	Allocation to an error category, either "OK", "Warning" or "Error"
Status description	Detailed information on the status (see chap. Status description)
System	Name of the system in which the variable is used. If "Design" is specified as system the variable is used in the design area (as a property value or by the program)
Used in	Specifies whether the text is accessed as a property value or from the source code
Source group/Source/Source detail	Specifies further details as to where in the application the text is used

2.3 Tool Bar

The tool bar of the cross reference contains different functions:

Scan	Starts the comparison between project databank and application. Prior to the access all changes in the project should be saved.
Close all nodes	Closes the detail display in all displayed lines.
Open all nodes	Opens the detail display in all displayed lines.
Display errors	Filters the display in the table for errors. If the switch is active the detected errors are displayed.
Display warnings	Filters the display in the table for warnings.
Display OK	Filters the display in the table for the OK status.

2.4 Status Descriptions

The cross reference establishes different states on the variables and texts of the project. To every status named "Status" a category is firmly allocated. With the "Error" and "Messages" categories the display in the table can be filtered.

Status list for the variable reference

Status description	Description	Status
OK	No errors were found with the use of an item.	OK
Defined but not used	The variable was defined in the project databank. It is not used in the application. The definition is redundant, and can probably *) deleted from the project.	Warning
Used but not defined	The variable was used in the application but was not defined in the project databank. In the course of the application an error from the variable kernel is to be expected when the application tries for the first time to register the used variable.	Error
Use error (Bit number)	Access to a single bit of a variable will trigger an error as the data type does not support such a high bit number.	Error

Status list for the text reference

Status description	Description	Status
OK	No errors were found with the use of the text.	OK
Defined but not used	The text was defined but not accessed in the application. Probably *) the text can be deleted from the databank.	Warning
Used but not defined	The text was used in the application but was not defined in the project databank. In the course of the application a faulty text output is to be expected.	Error

*): In this context **probably** means: Not all code constructs are interpreted by the cross reference. If variable or text names are for example passed to the appropriate function (e.g. "VWGet" or "GetText") as String variables from a data source the cross reference does not trigger the access.

Example for an access of VWGet identified by the cross reference

```
VWGet("Ch1.Simulation.Dummy.SawTooth1.Value")
```

Example for an access of VWGet NOT identified by the cross reference

```
int i = 1;
```

```
VWGet("Ch1.Simulation.Dummy.SawTooth" + i.ToString() + ".Value")
```