PC-based systems

ELOP II Factory

New Features

Information for Upgrade
from Version 3.5 to 4.0

HIMA

HI 800 051 AEA

For more information see the documentation on the CD-ROM and on our web site at www.hima.de.

More information can be requested from:

HIMA Paul Hildebrandt GmbH + Co KG
Postfach 12 61
68777 Brühl

Tel: +49 (6202) 709 0
Fax: +49 (6202) 709 107

E-mail: info@hima.com

Contents

Welcome to ELOP II Factory Version 4.0

This ELOP II Factory documentation presents the new functions of version 4.0. The images ("screenshots") and information should help you to make yourself quickly familiar with the additional possibilities of the new version.

This document concentrates mainly on the new features. Not all details are mentioned here as otherwise the quality of this overview may suffer from that. Nevertheless you will find sufficient information to try the new functions as soon as you have read this documentation. As usual, a detailed documentation is provided in the ON-LINE help.

Please also observe section "Using Data of Earlier Versions" in the files READ1ST.HTM and README.HTM.

Conventions

Character Format	*Used For*
italic	references and emphasis
SMALL CAPITALS	keys of the keyboard If several keys have to be pressed at the same time, they are connected in the text by the character +. ***Example:*** ALT+S means that the ALT-KEY and the S-KEY must be pressed simultaneously.
()	including one or various parameters with function calls
`Source`	commands, options, parameters, source examples

`BEGIN`
`  .`
`  .`
`  .`
`END`
	skipped lines in source examples

...	following parameters with the same format

Character Format	*Used For*
[]	identifying optional parameters ***Example:*** `COMPILE [-Option1] [-Option2] PROJECT`
|	The parameter before or after this character must be entered.

How to Enable the Current Version in Your Hardlock

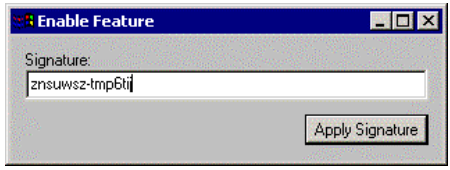
Enabling the version via signature

From version 3.5 onwards you must enable each new version in your hardlock!
The version is enabled by a "signature" (a character-digit-combination, also known as "**activation key**") which can be obtained from HIMA.

In case of a new delivery or if purchasing following-licenses you receive a hardlock in which the specific version has already been enabled by HIMA.

If upgrading to a version which is released later, you can enable that version yourself as follows:

1 Install the ELOP II Factory-base system.
2 Attach the hardlock to your computer.
3 Have the signature ("activation key") for the new version ready.
4 Start the Project Management.
5 A message informs you that the current version is not enabled in your hardlock.
6 Confirm this message (by clicking *OK*).
7 A dialog opens in which you can enter the signature:



8 Click *Apply Signature.*
9 A program message informs you about the successful enabling of the additional product. Confirm it, too.
10 Close the dialog by clicking  .
11 The Project Management is started.

Note Use this dialog to enable additional products directly in the Project Management:

1 Start the Project Management.

2 Open the menu *Tools* of the menu bar.
3 Select *Enable Feature*.
4 The dialog *Enable Feature* is opened: The remaining operation is analogous to the above description.

Additional info... can be found in the ON-LINE help under "How to Enable an Additional Product in Your Hardlock" (index "Enabling").

New Installation

You will notice at once that ELOP II Factory comes with a new installation program in version 4.0.
This change does not essentially affect you as ELOP II Factory-user. However, you may wish to inform yourself about the differences.

Both Language Versions are Installed

Now both language versions of ELOP II Factory are installed.
The "Control Center" offers a "language switch" after the installation so that you can easily switch the language of the user interface.

Please observe: You will only be able to use both languages, if your hardlock/softlock has been enabled for both languages.

Improved "Control Center"

The "Control Center" from where you start/call several functions/information for ELOP II Factory has also been reworked. You will find it easier to handle.

Changed Location of Program Files

The location of ELOP II Factory program files has changed (due to the new installation).

As a ELOP II Factory-user, free of special administrative tasks, you will not need to know the exact position of the program files. So HIMA does not provide a file list (not even in the ON-LINE help).
However, the ON-LINE help has been updated in those sections where information on a file position is given.

CHAPTER 1

Conversion

"What happens with my old data?"
This is the question which often passes through your mind, if you are using new versions.

Of course, you can use your data in the new version, but you must convert it.
Here you are informed about the necessary steps.

Please do read this chapter!

Additional Tool LCConvVer for Conversion

Please use the additional tool LCConvVer to update your data to the current level!
By using this additional tool you convert objects of the previous to the current version (e.g. version 3.5 to version 4.0).

You may know about its usage already from previous versions. If not, details on LCConvVer can be found in the HTML-manual included in the delivery. Open it as follows:

1 Start a "ELOP II Factory command prompt" (*Start*, *Programs*, ELOP II Factory).
2 Enter the following command: `LCConvVer -m`
3 The HTML-documentation for LCConvVer is opened.

Why an additional tool?

- Using the additional tool, the conversion cannot be done by mistake. If you have converted data, you cannot use it any longer in older versions.
  Reason: There is no downward compatibility between the versions.
- The conversion is performed centrally by this additional tool.
- The additional tool guarantees that the data is converted completely. Otherwise occurring errors might be recognized too late.
- Not every user of ELOP II Factory should be allowed to convert data.

Additional info... can be found in the ON-LINE help under "Why Conversion?" (index "Convert").

Notes on the additional tool LCConvVer:

- If you wish to convert objects from an older to the current version, you must execute the conversion for every version in between.
  Exception: Objects from version 3.5 on can be converted directly to version 4.0 – by using the conversion tool LCConvVer!
- The following data is not converted automatically by LCConvVer:
  - archives of the version management
  - archives (backup copies), created by command *Archive...*

  If you restore a project not already converted to the current version, you can have it converted directly from within the Project Management: Answer *Yes* to the prompt; *Do you wish to convert to the current version now?*

- When using the additional tool LCConvVer, you can create a backup copy of each object to be converted. In case of successful conversion this backup copy will be deleted resp. in case of conversion errors the objects will be restored using this backup copy.

Survey: Use Data of Former Versions

The ON-LINE help contains a table answering the following questions:

- Which ELOP II Factory-version must be converted?
- Which conversion tool must be used?

How to find this table:

1 Start the ON-LINE help for ELOP II Factory.
2 Go to tab *Index*.
3 Enter "Convert".
4 Double-click the index-entry "Convert".
5 Double-click the topic "Convert: Update to the New Version" in the new window.
6 Scroll downwards in the topic until the table appears.

Please observe that there is no downward compatibility.
If you modify (and save) data e.g. in version 4.0, you cannot use it any longer in earlier versions.

CHAPTER 2

Project Management

Version 4.0 provides again more comfort and better usability in the Project Management.

Different Identification for Program/Type Instance

The icon (in the structure window) makes it easy for you to recognize now whether the object is a program instance or a type instance:

The icon of a program instance is filled with color "magenta".

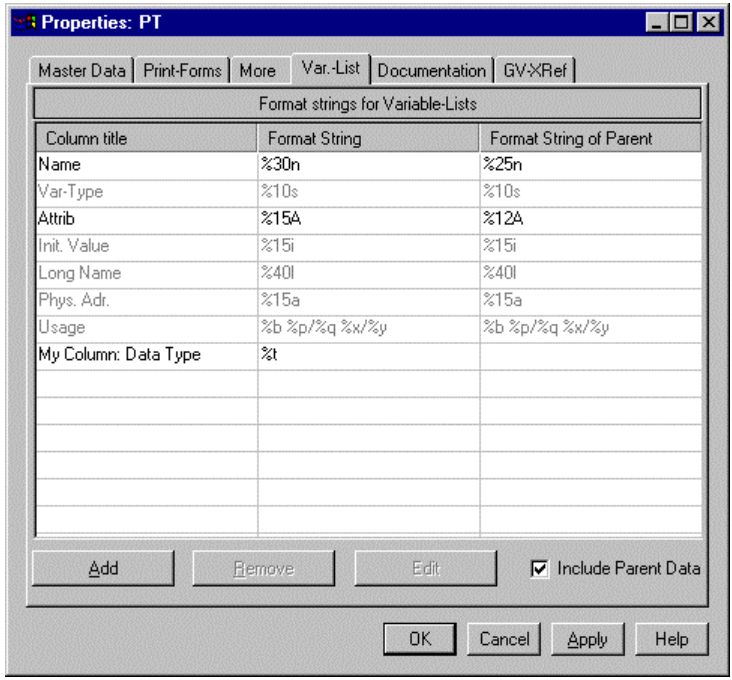
The icon of a type instance is filled with color "white"

Additional info... on both object types can be found in the ON-LINE help: e.g. index "Type Instance"

Including and Modifying Parent Data in Properties of Object

In version 4.0 some tabs in the properties of the object (pop-up menu of object, *Properties*) have been reworked.

Similar to "Inherit" The check box *Include Parent Data* offers new functionality in those modified tabs:



Adding Create items in these tabs as follows:

- Create local data items as usual, e.g. by clicking button *Add*.
  Local data items are displayed in black font color.
- Check *Include Parent Data*: The data items of the parent object are added.
  These parent data items are displayed in gray font color and handled like the local data items.
- Uncheck *Include Parent Data*: The data items of the parent object are removed, the local data items are kept.

Editing Now you can modify all data items in the tabs in a simple way. Double-click e.g. a field of the data item, enter the new text and press the ENTER-KEY.

Also new in the tabs is the pop-up menu by which you can manipulate the data items.

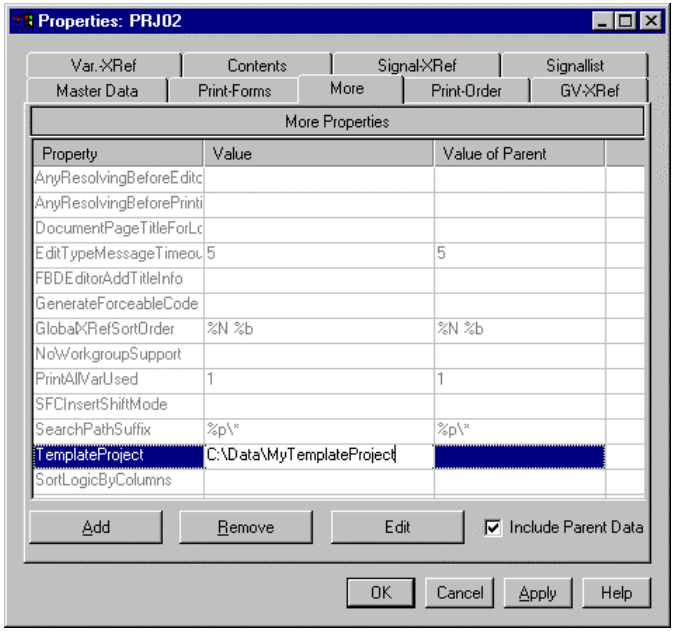
Additional info can be found in the ON-LINE help under "Creating, Deleting, Modifying Items in Tabs" (index "Parent Data").

Note The tab *Print-Def.* has been rename to *Master Data* in version 4.0.

User-Defined Template-Projects – As Many As You Like

In addition to the default template-project of HIMA you can create and specify your specific template-projects from version 4.0 onwards:

1 Copy the default template-project.
2 Adjust this copy to your requirements.
3 Define the property `TemplateProject` on tab control *More* and the project file as its value:



Note You can specify different template-projects in the properties
- of all folder objects and/or
- of the Project Management.

In principle the defined project is the template for projects/objects to be created anew. However, the behavior of ELOP II Factory might differ depending on where the property is defined.

Additional info... can be found in the ON-LINE help under "Default Templates for Objects" (index "Templates").

Path Mappings: Environment Variables are Allowed

For some time ELOP II Factory has already offered the possibility to manage path mappings centrally in the Project-Management-properties:

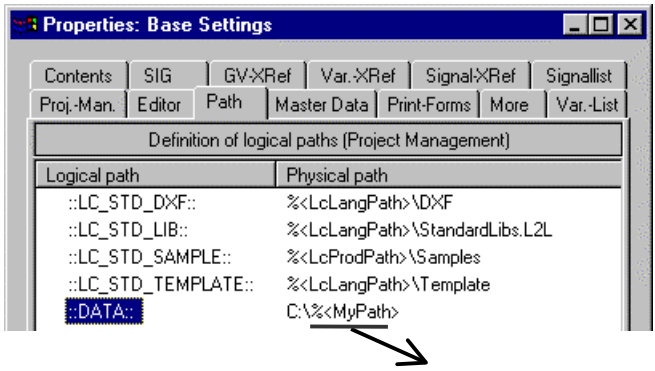
1 Define in tab *Path* (menu *Project, Properties*) a logical path and the appertaining physical path.
2 Subsequently you can use the logical path at other locations in ELOP II Factory (e.g. in tab *Print-Forms*).

Advantage Logical paths save work:
If the physical path changes, you only have to update the physical path at the central position in ELOP II Factory: on tab *Path*

Until now you could only enter the physical path as an absolute path.
Example: C:\MyData\ELOP II Factory\

New But from version 4.0 onwards you can integrate environment variables in the path name.
Syntax for naming an environment variable: `%<environment-variable>`

Example for integrating an environment variable in the physical path:



The logical path is resolved as `C:\MyData\ELOP II Factory`.
Condition: The value `MyData\ELOP II Factory` has been assigned to the environment variable `MyPath`.

If the environment variable does not exist, the place holder contained in the physical path will be resolved as empty string.

Several Definition Files for Project/Object Wizard

In advance: Operation and configuration of the project wizard and the object wizard are still the same. Only the administration of the offered templates has been changed.

Please recall The project wizard (under option *Create new project*) as well as the object wizard offers templates for creating project resp. objects.
Already before version 4.0 you could adapt the template list. For each wizard you had to change the contents of **one definition file**.

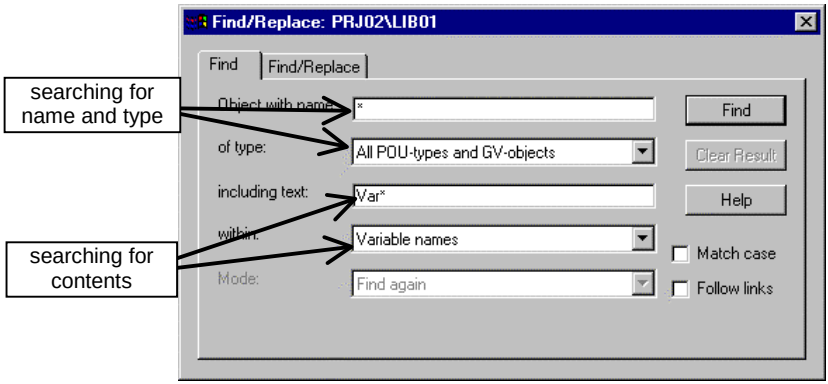
New Now the project wizard and the object wizard read the contents of **all definition files** existing in a certain directory.
Due to this modification it is not necessary any longer that you change the default definition file of HIMA but you may create your specific definition files.
Please observe: The location of the definition files has also been changed.

If you wish to offer your specific templates in the project wizard and object wizard, please read the detailed documentation in the ON-LINE help (to be found e.g. by index "Wizard").

Searching for POU-Name and POU-Contents

Now you can search in the Project Management for POU-names and/or data in POUs (e.g. variable data, instance names of blocks, comments etc.):

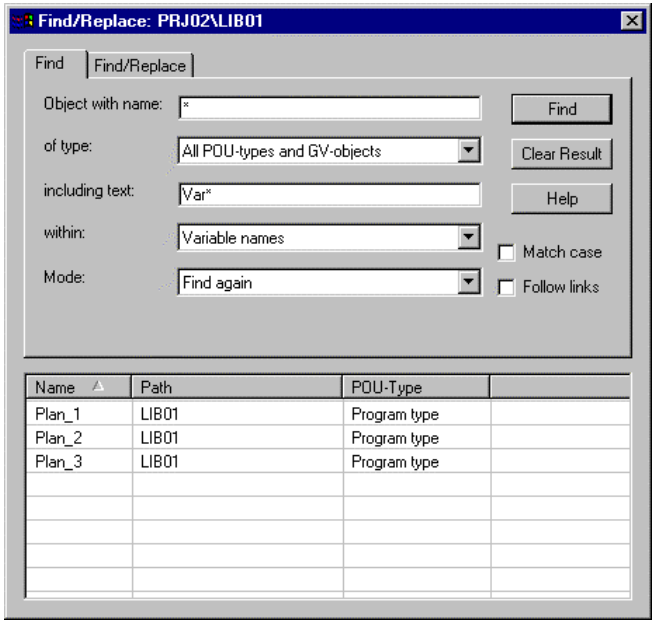
1 Open the pop-up menu of e.g. a project.
2 Select Find/Replace.
3 Define the required settings in tab *Find*.



Please observe: By default the global-variable objects are considered, too (by type "*All POU-types and GV-objects*").

4 Check *Follow links*, if links and objects behind the link should be considered when searching.
5 Click *Find*.

6 The search results are shown in the lower part of the dialog. Click a column header in order to sort the search results:



TIP: Double-clicking a found object opens the appertaining editor.

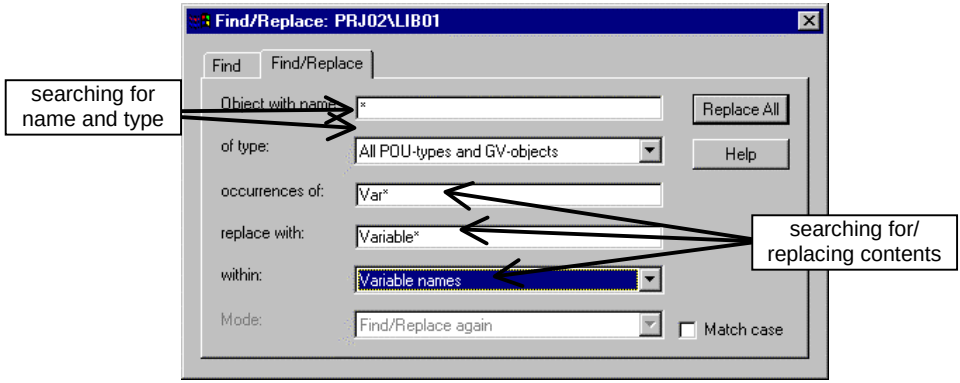
Note You can control extent and contents of the recent find results by the *Mode*-settings
Additional information can be found in the ON-LINE help under "Global Finding and Replacing" (index "POU, Finding").

Replacing Text-Contents of POUs

Similar to "Searching for POU-contents" you can replace data in POUs (such as variable data, instance names of blocks, comments etc.) in the Project Management:

1 Open the pop-up menu of e.g. a project.
2 Select *Find/Replace.*

3 Go to tab *Find/Replace*.
4 Define the required settings in tab *Find/Replace*.



Please observe: By default the global-variable objects are considered, too (by type "*All POU-types and GV-objects*").

5 Check *Follow links*, if links and objects behind the link should be considered when searching.
6 Click *Replace All*.
Please observe: Links and objects behind links residing outside the current project will not be considered when replacing.

What to replace? This is a simple way to replace variable names, instance names of blocks, comments or connectors.

TIP Before replacing you might wish to get a grasp in which POUs the texts will be replaced. This is how to list the search results and replace the texts in them:

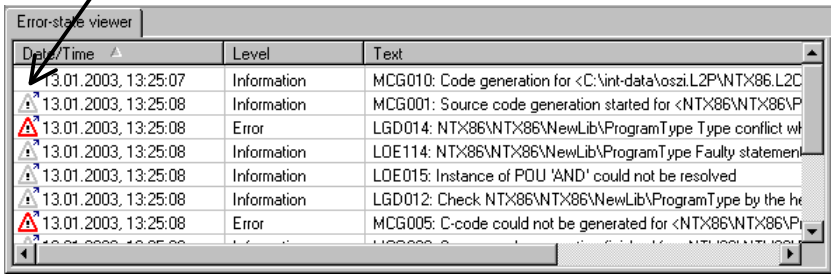
1 First search for POU-names and POU-contents in tab *Find*.
2 Transfer the search results by changing to tab Find/Replace.
3 Enter the search and replacement criteria.
4 Select the mode Find/Replace in results.
5 Click Replace All.

Object-Archives are Automatically Converted

Object-archives from previous versions will be automatically converted after restoring.
Up to now you had to convert the restored object yourself using the additional tool LCConvVer. This working step is no longer necessary!

Going from Error-State Viewer to Error Source

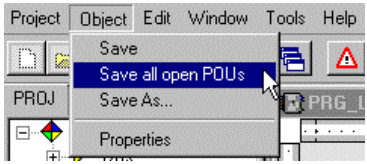
You will notice that some messages in the error-state viewer are marked with a "go-to"-icon.



Double-click such a message, and the object containing the reported error will be opened automatically. You can fix the error at once.

Saving All Opened POUs

Meeting ELOP II Factory user requests, the menu *Object* now offers a new command to save the modifications of all opened POUs at the same time:



CHAPTER 3

Editors

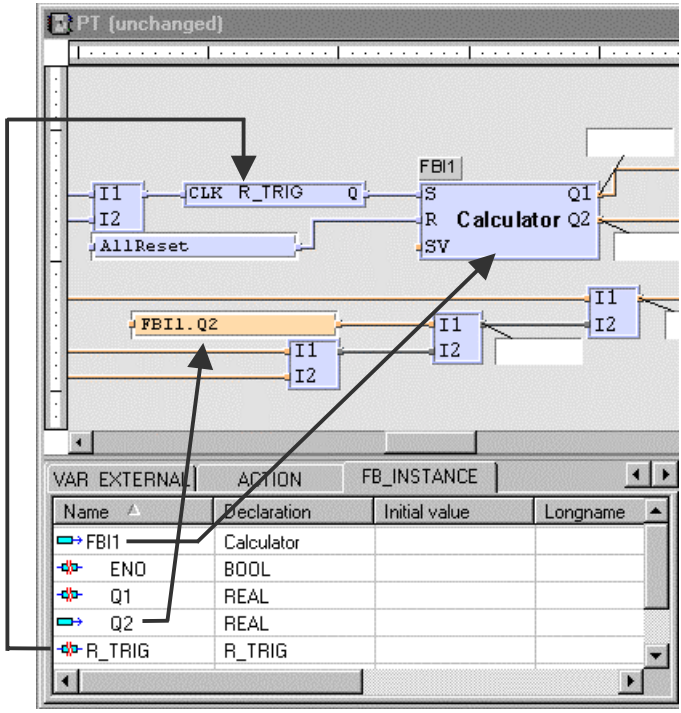
In version 4.0 the handling of the other editors has also been made easier and has been enhanced.

FB-Instances in Variable Declaration Editor: Also Available for FBD

For editing POUs Function blocks can be inserted in a POU via tab `FB_INSTANCE`.

For the FBD-editor this tab is a comfortable enhancement:
In the FBD-editor there is an existing feature: You can have an FB instance list created via plug-in (in the drawing field of the FBD-editor). This list offers default functions, such as *Sort*, *Filter*, *Print*.

But now you have all function block instances of a POU also closely at hand in the tab `FB_INSTANCE` of the variable declaration editor:



Presentation & operation

... of this tab is analogous to the other tabs of the variable declaration editor.
By means of the pop-up menus and their commands you can hide columns, filter/sort the dialog contents or go to the position of the FB-instance set in the drawing field (by *Go to occurrence...*).

Possibilities to modify

For your work with the FBD-editor there are some new and interesting features with the tab:

You can modify the instance name in the tab:

1 Select the name of the FB-Instance (= in column *Name*).
2 Click the name again.
3 Enter the new name and press the ENTER-KEY.
4 The instance name is automatically updated in the FB-instance already set.

Moreover, you can display the outputs of an FB-instance:

1 Open the pop-up menu for an FB-instance in tab `FB_INSTANCE`.
2 Select *Expand*.
3 The outputs of the FB-instance are indented (tree view).

TIP: You can drag outputs of the FB-instance into the drawing field. Hence, the created value field automatically displays the correct syntax for the access to the output.
Example: Output `Q2` of FB-instance `FBI1` is dragged into the drawing field. Contents of value field: `FBI1.Q2`

Printing behavior

When printing (pop-up menu of POU, *Print*) the FB-instances of tab `FB_INSTANCE` are not output in the variable list by default.
However, you can change this behavior by a property in tab *More*.

As up to now...

If you wish to e.g. modify or delete the FB-instance, do that at the set FB-instance itself: pop-up menu of FB-instance in drawing field, *Edit Type* or *Delete*

Additional info...

can be found in the ON-LINE help, e.g. under "Tab <FB_INSTANCE> Shows FB-Instances" (index "Function Block Instance").

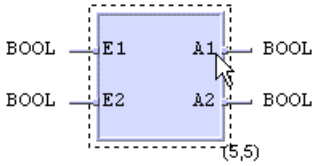
Improvements in Interface Declaration Editor

Version 4.0 offers some comfort features in the interface declaration editor.

How to Rename I/Os

New You are allowed to rename I/Os in the interface declaration editor:

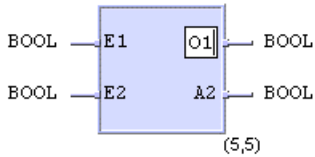
1 Point to the character next to the I/O, e.g. to "1" of output "A1":



2 Press and hold the ALT-KEY.

3 Click mouse button 1.

4 Enter the new text, e.g. "O1":



5 Exit the edit-mode by pressing the ENTER-KEY.

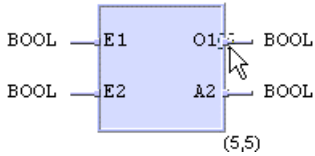
If instead of the I/O-names the alternative I/O-identifiers are displayed in the block image (pop-up menu for block image, *Alternate I/O-names* is checked), then renaming refers to those.

Known alternative However, you can still rename any I/O in the variable declaration editor: Select an existing declaration `VAR_INPUT` or `VAR_OUTPUT` (in the variable declaration editor), click it again, enter the new name and press the ENTER-KEY.

How to Move I/Os

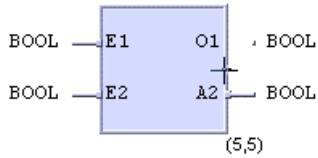
New You are allowed to move I/Os in the interface declaration editor:

1 Point to the I/O (the node on the block edge), e.g. the output "O1":

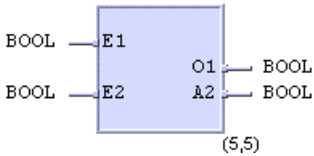


2 Press and hold mouse button 1.

3 Drag the I/O to the destination (a free position for an I/O on the block edge).



4 Release mouse button 1.



Known alternative However, you can still change the position of any I/O in the variable declaration editor:
Double-click an existing declaration `VAR_INPUT` or `VAR_OUTPUT` (in the variable declaration editor), in the dialog *Variable Declaration* enter a new value under *Position* and press *Update*.

Preview for New Comment/Value Field

If you created comment fields or value fields in the block image before version 4.0, ELOP II Factory positioned them at once in the block image. You had to move the field separately to the destination.

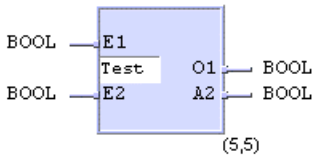
New Now ELOP II Factory offers you just the preview of the specific field at first and you can move the field to the destination at once.

Invisible Grid for Positioning

In the block image, objects (such as comment fields or value fields) can be created, moved, enlarged or reduced in size.
For positioning of these objects an invisible block image grid is available now. It helps you set objects more exactly by moving them "step by step" along the grid. Grid size is half the distance between I/Os.

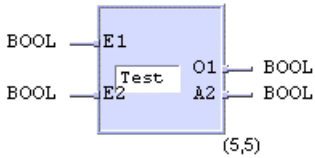
But if you want to move, enlarge, set an object "smoothly", you should press and keep pressed the ALT-KEY: You can position the objects without having to stick to grid points.
During the manipulation, you can see that the objects float instead of being moved almost "jerkily".

Example for positioning on grid: The comment field "Test" resides precisely between the inputs "E1" and "E2":



distance for grid of block image = half the distance between I/Os

Example for positioning independent of grid: The comment field "Test" resides more closely to input "E2":

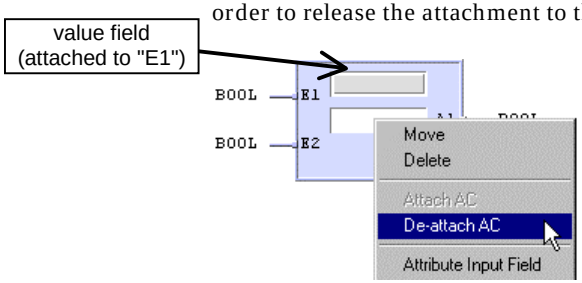


Entering Comments for Value Fields in the Interface

In the interface declaration editor it is possible to create comment fields which are attached to the interface (the block image): pop-up menu in interface declaration editor, *Create Comment Field*

From version 4.0 onwards you can attach such comment fields to value fields which are attached to an input of the block:

1 Open the pop-up menu for the comment field: Select *De-attach AC* (in order to release the attachment to the block image).



2 Open the pop-up menu for the comment field again: Select *Attach AC*.

3 Drag the mouse cursor to the value field and click mouse button 1.

4 In the comment field enter the required text and/or format strings.

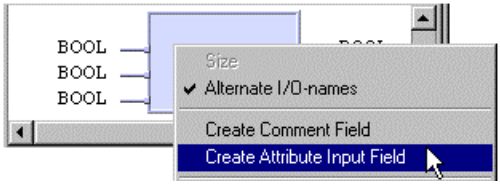
Please recall: You can create value fields for inputs as follows in the interface: pop-up menu for input, *Create Value Field*
Such value fields are necessary to define data of a block. The data can be changed when using the block in the drawing field.

Attribute Input Field Enables You Defining/Modifying Comment/Instance-Name for POU

The attribute input field has been available since the release of version 3.5.
However, it has been completed after production of "New Features for Version 3.5" and thus the description for version 3.5 could be found in file README.HTML only.
HIMA now presents this feature together with the new features of version 4.0.

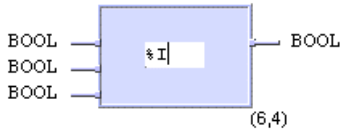
First Set the Attribute Input Field

1 Open the pop-up menu for the interface declaration editor in a POU.
2 Select *Create Attribute Input Field*.



3 Drag the preview of the attribute input field to the destination (in the block image).
4 Set the attribute input field by clicking mouse button 1.
5 Double-click the attribute input field.

6 Enter the format string `%c` (for comment) or `%I` (for instance name).



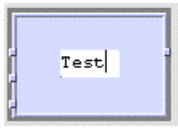
7 Save the POU: menu *Object*, *Save*

Effect of attribute input field If you now set this POU as an instance in the drawing field of another POU, you can define/modify comment or instance name for this instance.

Then Define/Modify the Field

1 Drag the POU from the structure window into the drawing field of another POU and drop it there.
2 The POU is set as an instance in the drawing field.
3 Double-click the attribute input field in the set instance.

4 Enter text in the attribute input field.

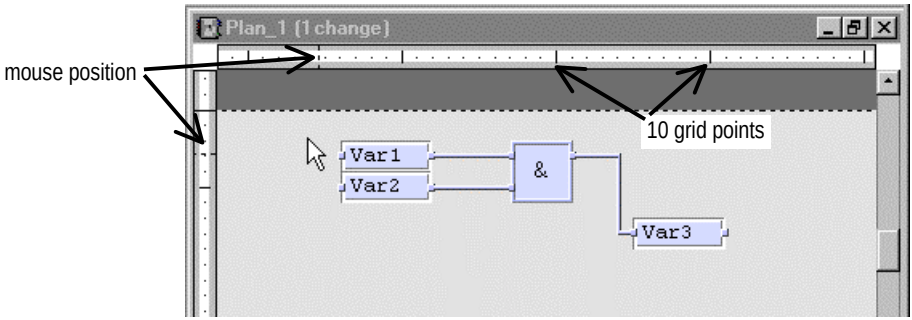


5 Click in the drawing field (outside the instance).
6 The text is accepted as comment (if format string `%c` has been entered in the interface declaration editor) or as instance name (if format string `%I` has been entered).

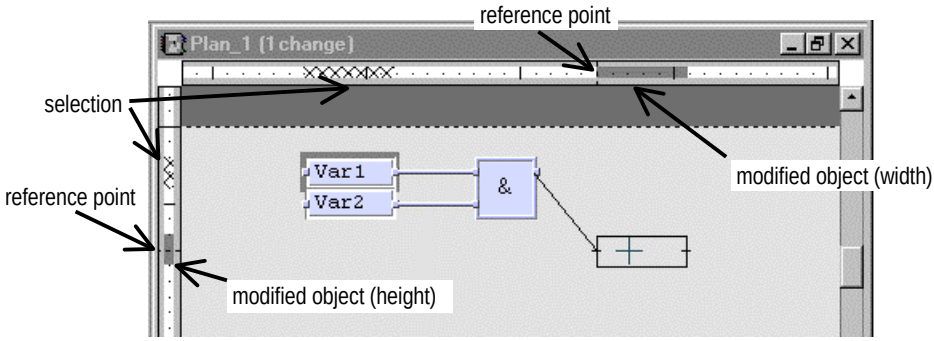
Additional info... can be found in the ON-LINE-help e.g. under "Defining Block Image" (index "Attribute Input Field").

Ruler in FBD-Editor Makes Positioning Easier

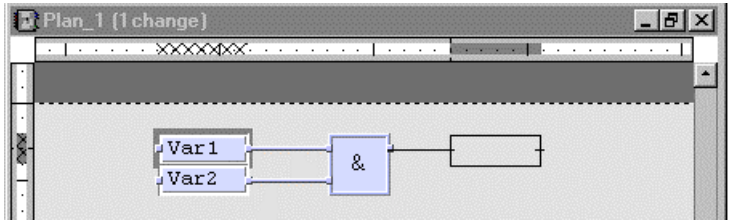
A ruler is displayed above and left of the drawing field in the FBD-editor. This ruler makes it easier to position the objects in the drawing field.



Use the ruler in order to align objects at the same height/width.
In this example the "modified" object (= preview of value field) should be moved to the same height as the selected value field "Var1":



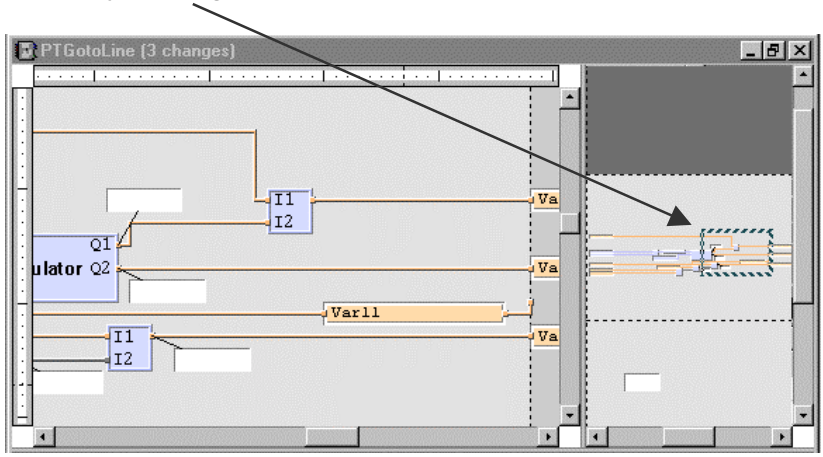
Move the preview of the value field upwards until the markings for "selection" and "modified object (height)" are positioned above each other:



Note on "reference point":
Usually the ruler indicates the current position of the mouse pointer within the page.
However, if you move/duplicate an object, the position of an object-"reference-point" is indicated. This position is e.g. the position of the sink node for value fields/connector fields or the position of the "upper" input for steps/transitions.

Improved Overview Makes Navigation within Page Easier

The area currently being edited in the drawing field of the FBD-editor is marked by a rectangle in the overview window:



HIMA has reworked the overview window slightly. You will get used to the comfortable functions once you have used them for the first time.

Dragging rectangle

Now you can drag the rectangle in the overview window and so change the displayed area in the drawing field very fast:

1 Press and hold mouse button 1.
2 Drag the rectangle in the overview window to the requested position: The displayed area in the drawing field is updated.
3 Release mouse button 1.

Vice versa, the rectangle in the overview window will move, too, if you scroll to a new position in the drawing field.

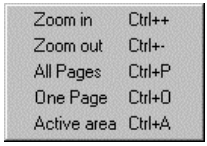
Modified behavior when clicking

If you click within a page in the overview, the focus is set on the corresponding spot within the current page of the drawing field.
But if you press and hold the SHIFT-KEY while clicking within a page, the focus will be the center of the page.

Before version 4.0 this behavior was the direct opposite.

Modified pop-up menu

In addition to these alternations you will find a modified pop-up menu for the overview window:



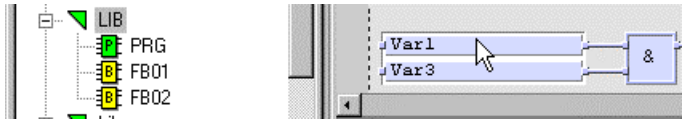
Detailed information about these operations can be found in the ON-LINE help (index "Overview Window").

"Auto-Scroll" of Drawing Field

If you wished to re-position an already set drawing field object outside the currently displayed drawing field before version 4.0, you had to scroll the drawing field to the new position yourself in order to set the object anew.
Now the drawing field is automatically scrolled, if movement operations are performed within the drawing field while holding the mouse button.

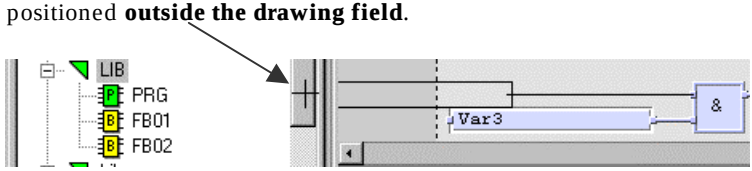
Example You wish to move a value field over the left edge of the drawing field:

1 Point to the value field.



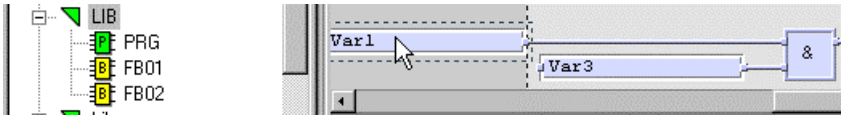
2 Press and hold mouse button 1.

3 Drag the preview of the value field to the left until the **mouse cursor** is positioned **outside the drawing field**.



4 The drawing field is automatically scrolled until you re-position the mouse cursor within the drawing field.

5 Drag the preview of the value field to the destination and release mouse button 1.



Note The drawing field is always scrolled automatically, if you perform these "movement operations" within the drawing field:

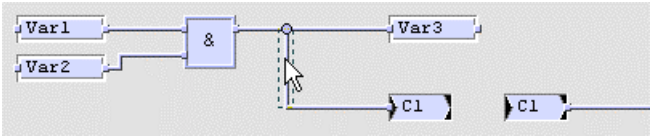
- moving drawing field objects
- copying drawing field objects
- drawing lines
- expanding step sequences
- re-assigning comment fields (with held modifier keys CTRL+SHIFT)
- creating sink/source connectors (with held modifier keys CTRL resp. CTRL+SHIFT)

Condition: You must press and hold the mouse button during the movement operation and position the mouse cursor outside the drawing field.

Improved Line Selection in FBD-Editor

As an alternative to command *Select Line* (e.g. in pop-up menu of a line) now you can select the complete line by this shortcut:

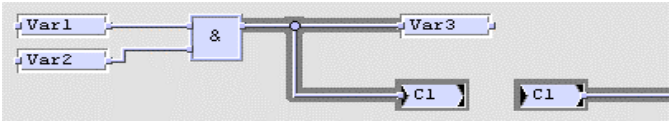
1 Point to the line (or one of its nodes):



2 Press and hold the SHIFT-KEY.

3 Double-click.

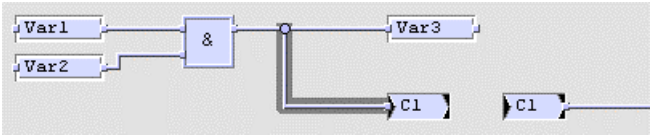
4 The complete line connection and possibly connected connector fields are selected:



Variant If you wish to select the current line part only:

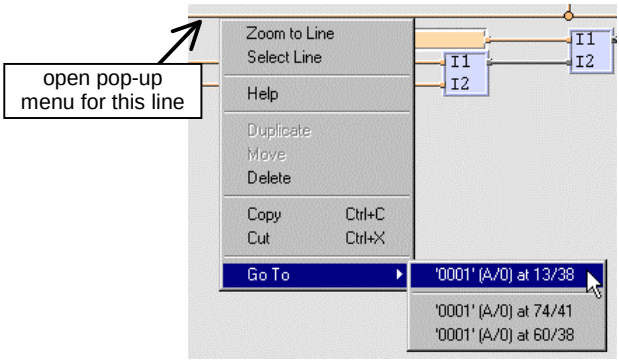
1 Double-click the line part but do not hold the SHIFT-KEY.

2 The line connection until the next fixed or locked node is selected:



"Go To" Facilitates Positioning onto Source/Sinks

You will find the new command *Go To* in the pop-up menu for lines. Use it to jump to the line source or one of the line sinks.



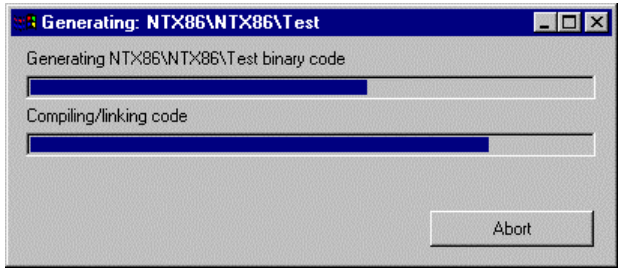
CHAPTER 4

Additional Products

Version 4.0 offers new possibilities in OFF-LINE simulation and documents-management.

Progress Bar/Abort for OLS-Initialization

If you start the OFF-LINE simulation (e.g. pop-up menu for resource, *OFF-LINE Simulation*), information for the simulation will be created based on the POU-data. The POUs determine how long this procedure will take.
A progress bar will give you feedback on the state of the code generation:



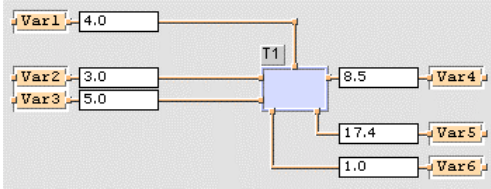
OLS can be aborted

You can also use the new progress-bar-dialog to abort the code generation – and thus the OFF-LINE simulation – at the earliest possible time: Click *Abort*.

OFFLINE-Simulation: Displaying Variable Values

From version 4.0 onwards you can display/hide the current values in the OFF-LINE simulation:

1 Open the OFF-LINE simulation.
2 Double-click an instance(FBD).
The FBD-editor in OLS-mode opens in the working area.
3 Now click  (*Display/Hide Values*) in the toolbar.
4 The current variable values are displayed beside value fields:



5 Click  again in order to hide the variable values.

Note You cannot "force" (= modify) those variable values, you have to force the variable value in the value field itself.

Printing Values from OFFLINE-Simulation

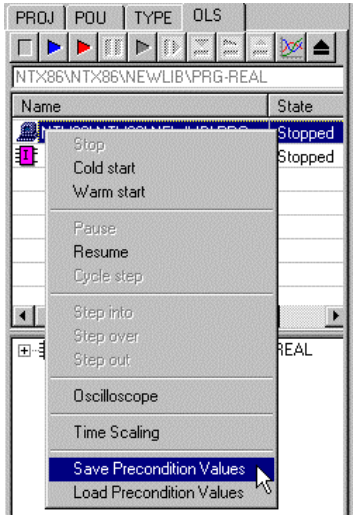
Before version 4.0 a function block diagram in OFF-LINE simulation was printed without values. Now you can print all displayed values from a running OFF-LINE simulation:

1 Start the OFF-LINE simulation and double-click an instance.
2 Open the pop-up menu for the drawing field (of the FBD-editor).
3 Point to *Print*.
4 Select Current View.
5 The current view of the function block diagram is printed with all values (up-to-date at the moment the command is selected).

Note OLT-fields (and their displayed values) will always be printed when the printout is started in the OFFLINE-simulation; compare "New Settings for Configuring FBD-Printouts, Do not Print OLT-Fields "

Defining Precondition Values for Simulation

You will find 2 new commands in the OFF-LINE simulation. Use them to save data of the OFF-LINE simulation and to continue later using those values: *Save Precondition Values* and *Load Precondition Values*



Procedure How to save/load precondition values in the OFF-LINE simulation:

1 If required, perform a first cold start (for initializing the entry points) and force the values in the instances.

2 Click (*Stop*) in the toolbar.

3 Open the pop-up menu of the system, select Save Precondition Values or Load Precondition Values.

4 In the dialog enter the file name resp. select a file.

5 Click (*Resume*) in the toolbar to resume the simulation.

Note Do not modify a saved file with precondition values! Otherwise it might not be possible to load the file any more.
Furthermore, you must not e.g. add or delete variables/instances/structure-elements. Because of that the data layout would be changed between saving and loading the precondition values. However, if such a modification is necessary nevertheless, create a new file with precondition values. You will be able to load this file.

Additional info... can be found in the ON-LINE help under "Defining Precondition Values for Simulation" (index "OFF-LINE Simulation, Defining Precondition Values ").

Improved Entry of REAL-Values

When entering REAL-values ELOP II Factory now accepts integer constants, too. ELOP II Factory will automatically convert such constants into REAL-values.

Example Integer constant "4" is automatically converted to REAL-constant "4.0".

This improved entry feature is offered with the following actions:

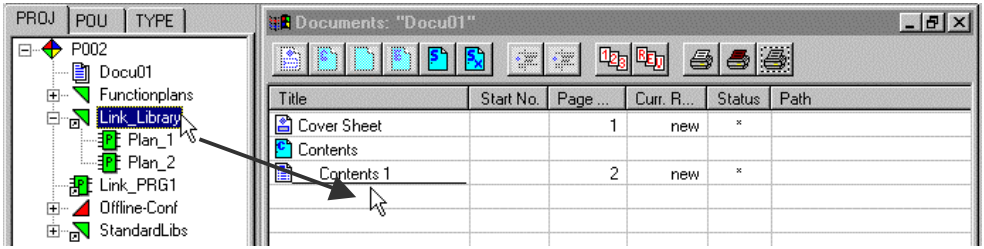
- when forcing REAL-values (in the OFF-LINE simulation)
- when assigning initial values for REAL-variables/-data-types (in variable declaration editor and data type editor)

Documents-Editor: Inserting Project-External Objects

In the documents-editor of ELOP II Factory it is now possible to insert objects residing outside the current project structure:

1 In the current project do create a link to the "external" object.

2 Drag this link into the opened documents-editor (and drop it there):



A link is principally represented in the documents-editor like the object to which the link points:

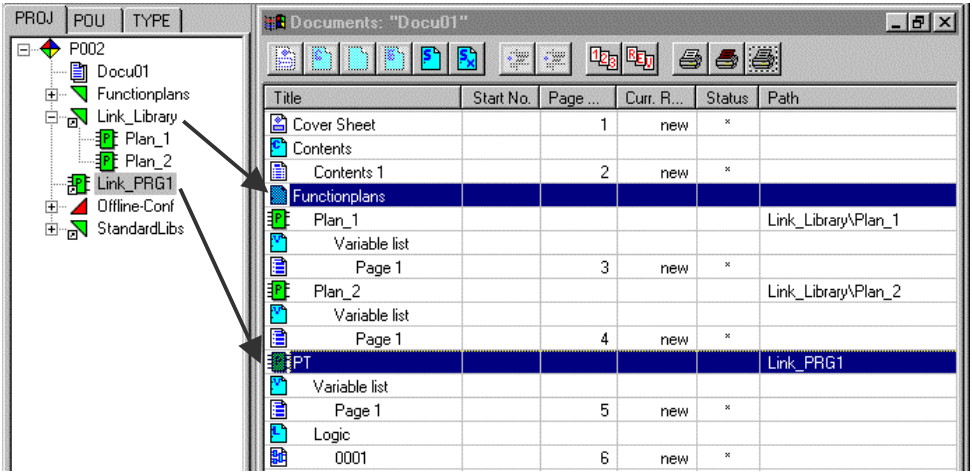
- In case of a link pointing to a library, the objects "behind" the link are also inserted.
  Example: Drag link "Link_Library" into the documents-editor. POUs "Plan_1" and "Plan_2" behind the link are inserted at once.

These POUs can be identified in the documents-editor by path "`Link_Library\Plan_1`" and "`Link_Library\Plan_2`" (= name of link and link-relative path).

- In case of a link pointing to a POU, only the POU is inserted.
  Example: Drag link "Link_PRG1" into the documents-editor. Only the referred POU "PT" is inserted.
  This POU can be identified in the documents-editor by path "`Link_PRG1`" (= name of link).

Restriction: Links in objects that are dragged into the documents-editor are still not considered.
Example: A folder contains a link to the standard library. Drag the folder into the documents-editor. The standard library is not inserted in the documents-editor.

Example for the documents-editor after updating the table of contents:



Additional info... The sections (e.g. Variable list) and pages (e.g. Page 1) associated to the objects are automatically inserted after updating the table of contents. on the documents-editor can be found in the ON-LINE help: index "Documents-Editor"

Checking Object-Integrity

Errors in ELOP II Factory-objects might occur in rare cases due to unpredictable reasons. That would lead to inconsistencies in the object data which might not be recognized before visually controlling the object (e.g. opening the POU in the POU-editor), before generating code or before ONLINE-operation.

You can use the tool LCCheckObj can be used to check the integrity of the ELOP II Factory-objects and to detect such errors earlier.

Details on LCCheckObj can be found in the HTML-manual included in the delivery. Open it as follows:

1 Start a "ELOP II Factory Command Prompt" (Start, Programs, ELOP II Factory).
2 Enter the following command: `LCCheckObj –m`
3 The HTML-manual for LCCheckObj is opened.

Exporting/Importing Path Mappings: New Options of LCExpSet/LCImpSet

If you wish to re-use properties already set for a ELOP II Factory object, the additional tools LCExpSet und LCImpSet already offered the required features before version 4.0.

From version 4.0 onwards you can now also export the path mappings (properties of Project Management, tab *Path*) by using LCExpSet. Then those path mappings can be re-imported by LCImpSet.
That is a comfortable way to transfer path mappings e.g. into another ELOP II Factory-installation.

Examples for syntax of the new option of LCExpset/LCImpSet (in a "ELOP II Factory Command Prompt"):

```
LCExpSet --pathmap >Paths.csv
LCImpSet --basesetting Paths.csv
```

Additional info... on using the additional tools can be found in the HTML-manual included in the delivery. Open it as follows:

1 Start a "ELOP II Factory Command Prompt" (*Start*, *Programs*, *ELOP II Factory*).
2 Enter the following command: `LCExpSet -m` or `LCImpSet -m`
3 The HTML-manual is opened.

HIMA Paul Hildebrandt GmbH + Co KG
Industrie-Automatisierung
Postfach 1261 68777 Brühl
Telefon: (06202) 709-0 Telefax: (06202) 709-107
E-Mail: info@hima.com Internet: www.hima.de