



COPADATA
do it your way

zenon manual

Distributed engineering

v.7.20





©2015 Ing. Punzenberger COPA-DATA GmbH

All rights reserved.

Distribution and/or reproduction of this document or parts thereof in any form are permitted solely with the written permission of the company COPA-DATA. Technical data is only used for product description and are not guaranteed qualities in the legal sense. Subject to change, technical or otherwise.

Contents

1. Welcome to COPA-DATA help	5
2. Distributed engineering	6
3. Information about SQL Server	7
4. Possibilities of the multi-user administration.....	7
4.1 Local Server	7
4.2 Local server to which several clients access	8
4.3 Access server with multiple clients	8
5. Creating a new multi-user project	9
6. Convert single-user projects to multi-user projects	11
7. Converting multi-user project to single user project.....	12
8. Working with a multi-user project.....	13
8.1 Extend and synchronize project.....	13
8.2 Changes to existing elements	14
8.3 Deleting projects	15
8.4 Disconnect client.....	16
8.5 Backup mechanism	17
8.5.1 Project backup.....	17
8.5.2 Restore project backup	18
8.6 Allow, accept and reject changes.....	18
8.6.1 Status of the elements and the modules	20
8.6.2 Key of the symbols	22
8.7 Deleting elements	23
8.8 Synchronization of the multi-user project	24
8.9 Recipegroup Manager.....	24
8.10 VBA and VSTA.....	25
8.11 Close the Editor.....	25
9. Working offline	26

9.1	Switch offline	27
9.1.1	Elements/modules are locked by another user	27
9.1.2	Elements/Modules are checked ot by the same user during offline switch	27
9.1.3	No elements/modules are checked out during going offline.....	27
9.2	From offline to online	28
9.3	Possibilities in offline operation.....	28

1. Welcome to COPA-DATA help

GENERAL HELP

If you cannot find any information you require in this help chapter or can think of anything that you would like added, please send an email to documentation@copadata.com (<mailto:documentation@copadata.com>).

PROJECT SUPPORT

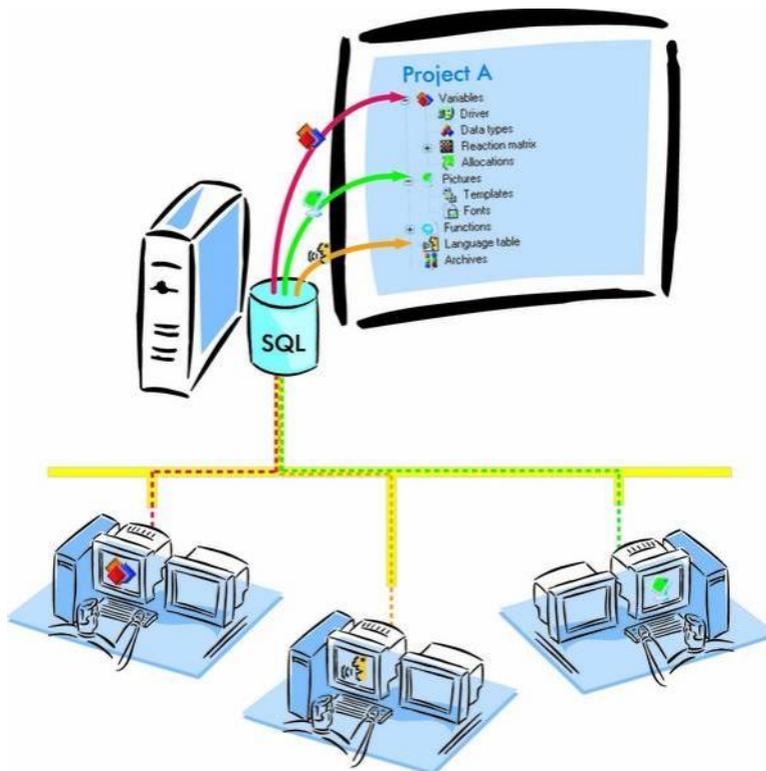
You can receive support for any real project you may have from our Support Team, who you can contact via email at support@copadata.com (<mailto:support@copadata.com>).

LICENSES AND MODULES

If you find that you need other modules or licenses, our staff will be happy to help you. Email sales@copadata.com (<mailto:sales@copadata.com>).

2. Distributed engineering

The Multi-user-function of zenon makes it possible, that multiple users concurrently work at the same project. For this, the projects are stored on a central SQL Server (on page 7) and can be worked on from different workstations at the same time. The central storage of the project will be called **server DB** in the following text.



For example: One user creates the screens while another one can concurrently define the variables.



Information

Hint for terminology:

"Check in" equals property **Accept changes**

"Check out" equals property **Enable changes**

IP PROTOCOL

The communication in the zenon network can be done via IPv4 or IPv6. For details see the Network manual, Requirements chapter.



License information

Part of the standard license of the Editor and Runtime.

3. Information about SQL Server

ZENON FROM VERSION 7.00 SP0 ON

For distributed engineering all participants access a central SQL Server. From zenon version 7.00 SP0 on all participants must use version 7.00 or higher:

- ▶ Editor: from 7.00 SP0 on
- ▶ DBSrv: from 7.00
- ▶ SQL Server: SQL Server 2008 R2 Express Edition

UPDATE

When updating the project server to SQL Server 2008 Express, all Clients must also be updated. SQL Server 2005 and SQL Server 2008 are not compatible.

4. Possibilities of the multi-user administration

The multi-user administration can be used for:

- ▶ Standalone (on page 7) in order to test changes without risk
- ▶ a local Server (on page 8) with several clients
- ▶ a network Server (on page 8) with several clients

4.1 Local Server

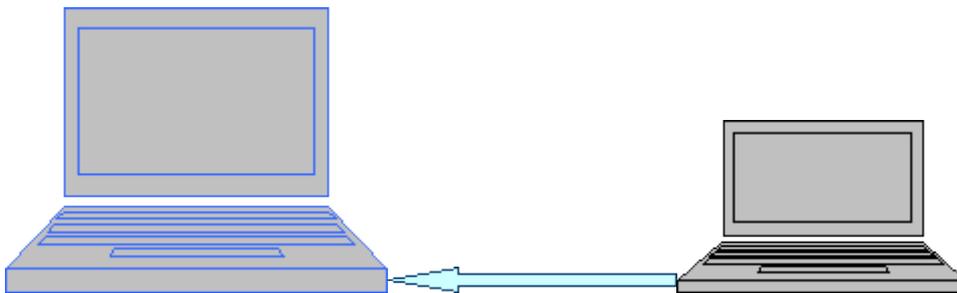
In this case the server project AND a local project copy are stored at the same PC.

It is possible to create multi-user projects on one single computer. This makes for example sense when you want to check changes first. To do this e.g. changes are allowed for a screen, it is edited and tested

in the Runtime. After this the changes can be checked in or rejected. The original screen stays unchanged, until the changes in the screen are accepted.

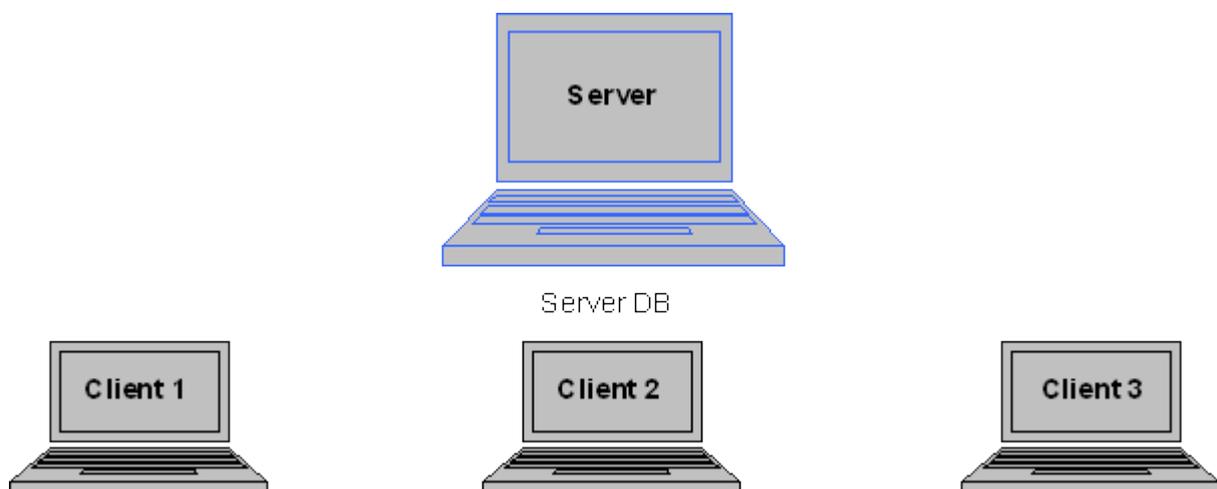
4.2 Local several to which several clients access

The server database and a local project copy are on one PC. The clients also work with a local project copy. Changes in the local project copy are transferred to the server database, when the changes are accepted. On the server computer also a local project copy is edited.



4.3 Access server with multiple clients

The server database is on the central computer. Multiple Clients connect to this server. Local project copies are created on each client.



Local project copies 1-3 of the Server DB



Information

A mixture of the described configurations is possible. E.g. each computer has one or more server databases for different projects and at the same time only one local project copy of another server database.

SERVER WITHOUT LOG INS

On servers on which no user is logged in, file `zendbsrv.exe` must run as device.

The `zendbsrv` service starts the `zensysrv` service for logging tasks. For the operating system Windows 7 this is not permitted via **Remote Transport** due to security reasons. The file must to be registered manually:

```
zendbsrv.exe -service
```

Hint: Also register file `zensysrv.exe` as service: `zensysrv.exe -service`



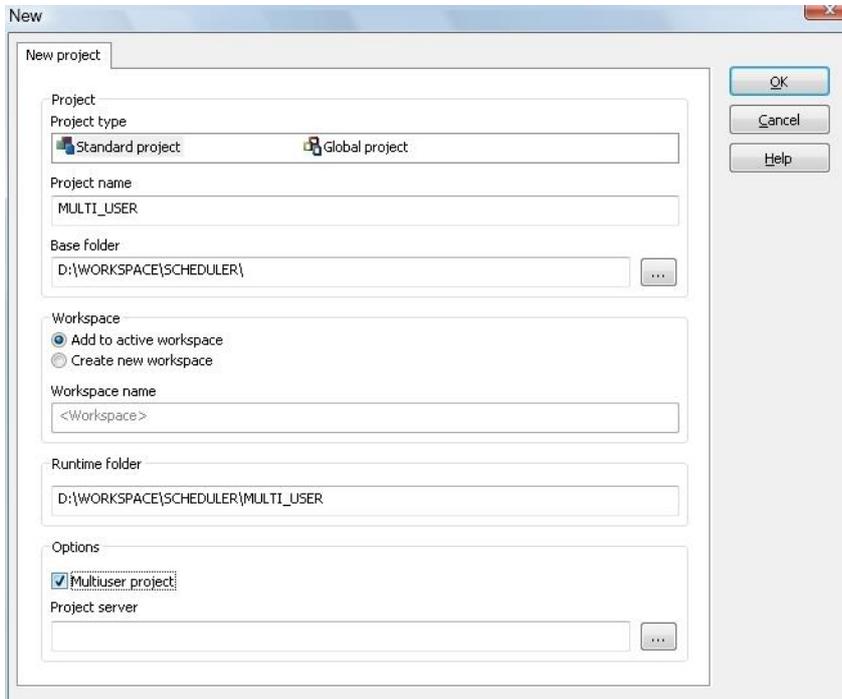
Information

*As soon as both files are registered, you can no longer use the **Remote Transport** under Windows 7, i.e. start the Runtime via **Remote Transport**.*

5. Creating a new multi-user project

When a new project is created, the checkbox **Multiuser project** has to be activated. Enter or select the server for the multiuser project in field **project server**. If a server is selected, the checkbox **Multiuser project** is automatically activated.

If no server is selected in field **project server**, the project, i.e. the server database, is created on the local PC. A local project copy is created automatically with which you can work directly in the Editor.



REQUIREMENTS:

- ▶ Project database server (zendbsrv.exe) must be started on the server and the client
- ▶ firewall setup:
 - you must unblock the programs zendbsrv.exe and sqlservr.exe.
 - SQL browser must be enabled for the Server and the Client.
- ▶ Necessary open ports:
 - 1103
 - 1433 for SQL Server: this port can change therefore the SQL Server process must be taken out completely from the firewall
 - 1434 for SQL Browser
- ▶ The name resolution in the network must work
- ▶ Reverse lookup must deliver the same computer name both at the server and at the client

 **Attention**

Workspaces do not support multiple users because they are station-specific. So the project hierarchy can be defined differently on different computers. The users have to communicate directly so that the hierarchy is the same on all stations.

CREATING A CLIENT FOR A MULTIUSER PROJECT

Execute the following steps in order to create a client for a multiuser project:

1. Select File -> Insert existing project...
2. In the following dialog select under **computer name** the name of the Server-DB. Under **Project selection** select again the computer name and the server name (DB)
3. Select `Multiuser project`

NOTES ON CONFIGURATION WITH ZENON LOGIC

Note when configuring with zenon Logic:

- ▶ In the zenon Logic Workbench, lines can be created in the **All projects** node **MultiSpy** and the global **Binding** Editor can be called up. The following is applicable for these settings: They are
 - Not saved as multi-user
 - Not transferred to the project server
 - Removed again in the next synchronization
- ▶ If new zenon Logic projects are created in the zenon Logic Workbench via File -> Add new project, the following is applicable: They are
 - Not transferred to the project server
 - Deleted again in the next synchronization

Attention: Multi-user zenon Logic projects can only be created using the zenon Editor.

6. Convert single-user projects to multi-user projects

To convert a single-user project to multi-user project:

1. Make sure that the global project is a multi-user project.
If this is not the case, remove the existing global project from the workspace and create a new multi-user global project via the context menu of the workspace. Project new

2. Then convert the single-user project: in the context menu select menu item Multi-user -> Convert to multi-user project
3. You receive a warning that you cannot undo this step.
You can however create (on page 12) a single-user project from a project backup of a multi-user project.
4. Confirm the warning by clicking on **Yes**.



Information

At the conversion the computer on which the action is carried out is defined as the database server for the project. This property cannot be changed later.

7. Converting multi-user project to single user project

To convert a multi-user project to single-user project:

1. Make sure that all project changes are taken over by all clients:
 - Project -> Context menu -> Multi-user -> Synchronize
2. Create a project backup:
 - Node project backup -> Context menu -> Create backup
 - Enter a name and a description in the dialog. You can also backup a local project copy.
3. Create a new project from the backup:
 - Workspace -> Context menu -> Restore project backup.
 - Select the desired project backup.
 - Activate checkbox **Create new project**.
 - Confirm your selection by clicking **OK**.
 - Enter a new project name.
4. The project is created as single-user project.
The multi-user project also still exists.

Note: To convert the local project copy to a single-user project, you can also select option Project -> Save as ... in the Editor.

8. Working with a multi-user project

Multi-user projects enable manifold access to modules. When engineering multi-user projects, especially take care for:

- ▶ the proceed at Enable changes and Accept changes (on page 18) and their handling for changes in several modules
- ▶ Deleting (on page 23) elements
- ▶ Synchronizing (on page 24) elements
- ▶ the exceptions of VBA (on page 25) and RGM (on page 24)

RENAME PROJECT

Projects can be renamed with the **Save as** command, because it is always the local project copy that is saved.

To save a server project under a new name:

1. Make a backup of the server project.
2. Select **Restore backup as a new project** in the context menu.
3. Name it as you wish.

The project is now available with a new name.

8.1 Extend and synchronize project

If a new element (e.g. a variable or a screen) has been created, this element first is only available locally and changes are automatically allowed.

If the changes are accepted, another client must synchronize so that the changed elements are also available on his computer.

If after creating a new element the changes are rejected, it is, as if this element never had existed.

The difference between a new variable and a new screen is, that after a synchronization the variable is not visible on the other computers, if the changes have not yet been accepted. After a synchronization the screen is visible as **under construction** on the other computers.

See also chapter Screens / Frame editor.

After another user has added or deleted an element, the project has to be synchronized in order to receive the changes. This is done with the entry **Project/Synchronize** in the context menu of the project.

The following configurations can be done in this context menu:

Command	Description
Multi-user/synchronize	The project is synchronized.
Multi-user/activate offline mode	The offline mode (on page 26) is activated. Changes take effect only locally and are not taken over centrally as long as the project remains offline.
Multi-user/activate online mode	The online mode (on page 28) is activated. If elements have been edited and taken over by other users in the mean time, they overwrite changes from the offline mode.
Update local version	The local copy is updated with the Server DB. The project is reloaded into the Editor. <i>Attention: Exception: The Elements/Modules, that are checked out by this user.</i>



Information

The elements that have been checked out by the same user, that has done the synchronization, will not be overwritten by the synchronization. The local changes are protected from overwritten by the check out status.

Exception:

User A checks out elements offline. User B checks out elements online. If user A reconnects to the server, the elements checked out by both users are overwritten with the data of user B. If from a team one member is out of office, we recommend not to change the project in this offline phase!

8.2 Changes to existing elements

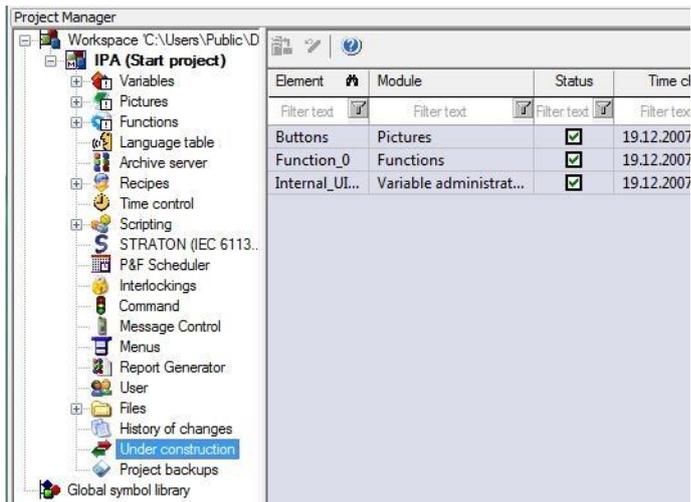
If changes should be done in an element, this element has to be checked out first. This prohibits that more than one user can access to the same element. Elements that are checked out cannot be changed by other users.

The Element can be changed by the user it has been checked out. This user has the exclusive change rights. For all other users this element is prohibited and locked.

In order to allow changes at an element, the user has to select "allow changes" in the context menu of the element. It is possible, to check out only single elements, e.g. one variable. To do this the entry

Allow changes in the context menu of the single variable in the detailview is activated. It is also possible to check out e.g. all the variables of the project at once via the node.

If a project is a multi-user project the entry **under construction** is added to the Project Manager. In the detail view of the list, the elements of the current project are listed. The default columns are: Status, Element, Target and Module. Additional columns can be added in the context menu of the column headers with the entry 'field selection'. With the additional columns the list shows an overview, which elements are edited by which user at which computer at the moment.



Information

If you want to use subprojects in a multi-user environment, make sure, that both projects have the entry Allow changes (on page 18) activated before dragging the subproject under the integration project.

An element/module can only be 'read' but not edited, as long as the user has not fetched the authorization for it. The authorization can be fetched with the context menu of the element **Allow changes** or directly at opening the element.

8.3 Deleting projects

If a project is deleted, only the local project copy is deleted. The server DB remains available and other clients can still work with it. It is still possible to add the project to the workspace. Here a new local project copy is created again.

If a user deletes his local project copy, all check out information and thus all changes of this user are lost.

8.4 Disconnect client

With the functionality disconnect client it is possible to disconnect the local project copy of an user from the server database. So all check out information of this user is deleted. Changing the elements locked by this user again is possible.

It can be necessary to disconnect a client, if the user is not in the office and has checked-out elements, which urgently have to be edited or the computer of this user is defective.

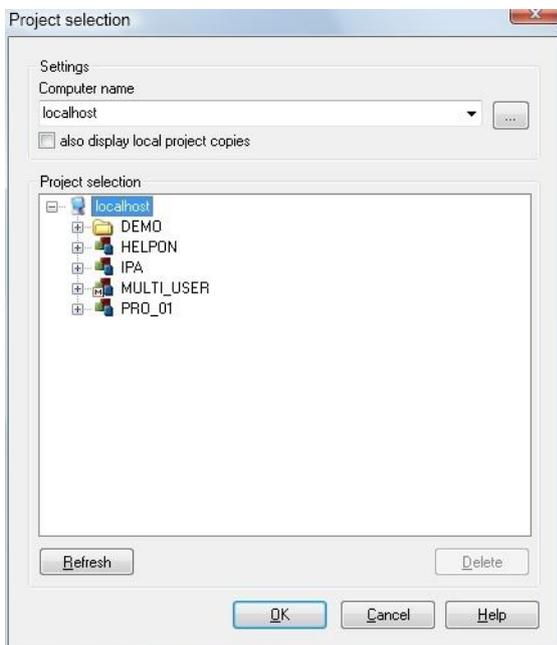
Attention

If the client of an user has been disconnected and the this user opens the project copy, the project is offline. It is no longer possible to go online. First the local project copy has to be deleted and again copied from the server.

REQUIREMENT:

A client can only be disconnected, if the computer, where the local project copy has been edited, cannot be reached.

With the main menu File -> Insert existing project the Project Database Manager can be opened. Under multi-user and connected local project copies you will find all users, which have (had) a local project copy.



With disconnect the entry for this user is deleted from the server database. After the client has been disconnected, the dialog can be closed.



Attention

This action cannot be undone! If the user on this computer again wants to access the multi-user project, the local project copy has to be deleted and again be inserted.

8.5 Backup mechanism

In general, the following should be noted for the backup mechanism with distributed engineering:

- ▶ Only local project states can be backed up.
- ▶ As long as they have not been synchronized, restoring is also possible.

8.5.1 Project backup

The project backup is executed in the project manager in the node **Project backups**.

Project backups of local project copies

A project backup can be made using the context menu of the **Project backups** module. To make a local project backup, the **Create local project copy** check box must be activated in the **save project backup** dialog box.

Local project backups can only be made within the same offline session as the backup.



Information

You can find further information in the project backup manual in the Project backup detail view tool bar chapter.

Project backup of server projects

A backup of a server project can be executed from each client, if the checkbox **Save local project** is not active.



Information

You can find further information in the project backup manual in the Back up and export project chapter.

8.5.2 Restore project backup

A project backup can be restored using the context menu of the **Project backups** module.

Local project restores can only be carried out within the same offline session as the backup.



Information

You can find further information in the project backup manual in the Project backup detail view tool bar chapter.

Restore project backup of local project copies

Restoring a local project copy is only possible, as long as no synchronization has been executed.

If a project has been synchronized since the last backup, the user only has the possibility to Restore backup as new project. Thus the project is restored as a single user project.



Information

You can find further information in the project backup manual in the Restore project backup chapter.

8.6 Allow, accept and reject changes

To make changes to a project, the respective elements (functions, variables, screens, ...) must be enabled for change (**Enable changes**). After the editing the elements must be protected from change and must be transferred to the Server (**Accept changes**).

Note: New and deleted elements are always transferred at the next synchronization. If you delete an element, it can still be edited on another computer. However it is finally deleted after the next synchronization.

Parameters	Description
Update local version	<p>If other users made changes in the server project, the local project copy can be updated with the menu entry.</p> <p>Updates only single existing Elements/Modules. New elements at the Server are therefore not available, synchronization is required.</p>
Enable changes	<p>In order to be able to change elements, the status 'Allow changes' has to be activated for them. This can be done in the context menu of the selected element. To make quicker work possible, you can enable changes for the whole module at variables. This means that all variables, drivers and data types are locked for all other users.</p>
Accept changes	<p>The changes are transferred into the server project. The element/module is then unlocked for other users.</p> <p>If changes were allowed for a complete module, they only can be accepted for the module.</p> <p>This means, that all variables, drivers and datatypes are transferred and unlocked for other users. Accepting a single element is not possible in this case.</p> <p>The changes in the local project copy are rejected and overwritten by the server data. The element/module is then unlocked for other users.</p>
Undo changes	<p>All variables, drivers and datatypes are rejected and unlocked for other users. Rejecting a single element is not possible in this case.</p>



Information

*When you change the project name, **Undo changes** is not possible. The changed project name remains.*

CHANGES ON SEVERAL MODULES AT THE SAME TIME

If a module is enabled for changes and if elements of other modules are newly created and linked with this module, all affected elements/modules must be set to **Accept changes**. Enable the newly created elements/modules first and then enable the already existing ones. If you now enable the module which was originally disabled, you may lose new engineering.

Example:

- ▶ On one Client the RGM is checked out.
- ▶ A new recipe group is created. Then a new recipe.
- ▶ An existing variable is added to the recipe group.

- ▶ Three new variables are created and added.
- ▶ Interlockings are checked out and a new interlocking is added.
- ▶ A new keyboard screen is created with a new frame.
- ▶ In the recipe group the new created interlocking is assigned.
- ▶ The keyboard screen is assigned.
- ▶ A new variable for the visibility is created and linked with a variable in the recipe group.

If only the recipe group is set to **Accept changes** and another Client is synchronized:

- ▶ the other Clients gets:
 - a recipe group with a linked variable
 - and a recipe which contains the variable
- ▶ all other information is lost

If this Client disables, edits and enables the recipe group, all newly engineered information is lost.

For correct acceptance **Accept changes** must be carried out for:

- ▶ RGM
- ▶ Recipe group
- ▶ Recipe
- ▶ Variables
- ▶ Interlockings
- ▶ Keyboard screen
- ▶ Frame for keyboard screen

8.6.1 Status of the elements and the modules

If an Element has been checked out, the Symbol in the project manager and the detail view changes. The user who has checked out the element gets a checkbox with a green check mark displayed. All other users get a checkbox with a red x. A ! means, that within the node single elements are checked out.

See chapter Legend (on page 22).

In the special case of the variable administration the symbols are displayed slightly transparently, if the module has been checked out.



Example

All variables are checked out by a user. Another user cannot change these variables, but he can add an additional variable.

Exception: Module checkout

The following actions are only possible if the whole module is set to **Enable changes**:

- ▶ creating array variables
- ▶ creating structure data types
- ▶ copying and pasting array and structure variables
- ▶ creating variables based on a structure data type with array
- ▶ changing the array settings of a variable
- ▶ changing array settings in zenon Logic of a variable with zenon profile



Information

*If a variable is created without the whole module **Variable** set to **Enable changes**, the dimensions are automatically set to 0. This prevents creating a structure variable unintentionally.*

The following elements/modules can only be set to **Enable changes** Or **Accepts changes** as a whole:

- ▶ Archives
- ▶ User
- ▶ Menus
- ▶ Reaction matrix
- ▶ Standard recipes
- ▶ Recipegroups
- ▶ Scripts
- ▶ Fonts
- ▶ Interlockings
- ▶ Allocations

DISPLAY CHANGED ON DATE IN THE DETAIL WINDOW

In the detail window there are two columns available which information about the last time changes were made:

- ▶ **Last change accepted**: Displays at all lists, whose content concern distributed engineering, the time of the last update in the database or of the last check in.

- ▶ Last modified: Displays time stamp of the last change of the files in the file system.

If the columns are not displayed, you can activate them via:

1. right-clicking on the column header
2. in the context menu select menu item **Column selection**
(It depends on the module which columns are available.)
3. select the desired columns in the selection list on the left-hand side (multi-select is possible)
4. add the columns by clicking button **Add** -> to the right-hand list
5. Via buttons Up and Down you select the desired position
6. confirm and close the dialog by clicking **OK**

8.6.2 Key of the symbols

The following symbols indicate the status of the elements in multiuser projects.

(for example: A screen can be changed by a user )

ELEMENTS IN THE DETAIL VIEW OF THE PROJECT MANAGER:

Symbol	Description
	New element has been added in the local project copy. On the first check-in the <code>element</code> is transferred to the server.
	New module has been added in the local project copy. On the first check-in the <code>module</code> is transferred to the server.
	Elements have been marked as deleted in the local project copy. On the first check-in the <code>element</code> is deleted on the server.
	Modul has been marked as deleted in the local project copy. On the first check-in the <code>module</code> is deleted on the server.
	Element can be edited by the current user.
	Module can be edited by the current user.
	Element is locked by another user.

DISPLAY IN THE PROJECT TREE:

Symbol	Description
	Module can be edited by the current user.
	Module is edited by another user and thus is locked for the current user.
	At least one element of the module is marked for editing (not the entire module).

8.7 Deleting elements

Data base elements are only displayed as deleted at first. At the other clients no more changes can be done to these elements, they are displayed as "locked". If „accept changes“ is selected, the elements are finally deleted. It is possible to undo the deletion with "undo changes".



Example

This is true for variables for instance: The changes have to be accepted to allow deleting.

File elements such as screens, files (language table, graphics) etc. are directly deleted when a confirmation is done on the executing computer. New and deleted elements are always transferred at the next synchronization. If you delete an element, it can still be edited on another computer. However it is finally deleted after the next synchronization.

8.8 Synchronization of the multi-user project

After another user has added or deleted an element, the project has to be synchronized in order to receive the changes. This is done in the context menu of the project with the entry Project -> Context menu -> Multi-user -> Synchronize or **All modules/update local version**.

The following configurations can be done in this context menu:

Parameters	Description
Synchronization	The local copy is updated with the Server DB. The project is reloaded into the Editor. <i>Attention:</i> Exception: The Elements/Modules, that are checked out by this user.
Update local version	See synchronize
Enable changes	Checks out Elements/Modules
Accept changes	Checks in Elements/Modules
Undo changes	The changes of the element are not accepted but discarded. After Undo changes the project is overwritten by the Server database again. <i>Attention:</i> If changes are undone at an element that has not yet been "accepted", the changes that depend on this element are also undone.

New and deleted elements are always transferred at the next synchronization.



Information

The Elements/Modules that have been checked out by the same user, that has done the synchronization, will not be included into the synchronization.

8.9 Recipegroup Manager

FILTER DIALOG IN THE RUNTIME

If you activate property **Show dialog in the Runtime** at the options of the Recipe value list in the screen switch, there are only restricted options available in Windows CE:

- ▶ Recipe Filter
- ▶ Column settings (for recipe table)

Recipe selection, Equipment modeling and column settings for the **recipe list** are not available.

8.10 VBA and VSTA

Select the entry **Open VBA Editors** or **Open VSTA Editor** in the context menu of **Programming interfaces**, then the Editor is checked out and changes are enabled.

Attention: In general in VBA changes are possible even if the element has been set to **Enable changes** on another computer.

Therefore consider the following before making changes via VBA:

1. before you take over changes: check whether you are dealing with a multi.user project
2. if yes: check whether the element is added by another Client
3. if yes: check whether the element has the local status **Enable changes** (check out)
 - a) if yes: the object can be changed,
 - b) if no: the object must be set to **Enable changes**
 - check again whether the object has the **Enable changes** status
 - if yes: the object can be changed in VBA
4. after the changes the object must be set to status **Accept changes** (check in) via VBA

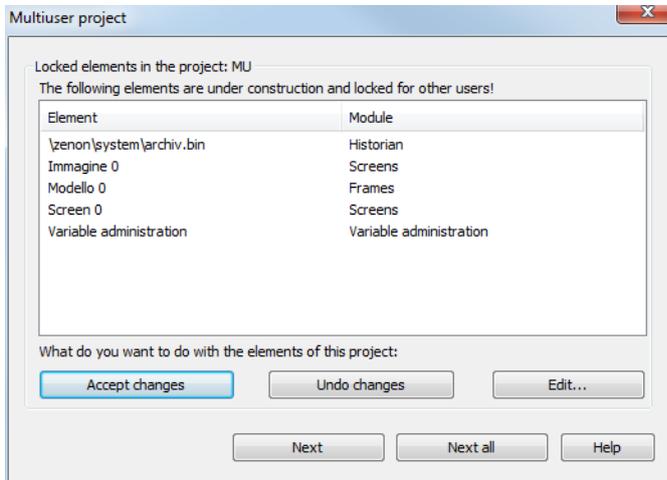
Attention

In multi user projects the VBA Editor must not be opened with Alt+F11. If VBA is not checked out, changes are not accepted. When the VBA Editor is opened, a warning message is displayed that indicates which projects are affected.

8.11 Close the Editor

Each user should accept the changes of his objects before closing the Editor, the workspace or the project. As long as this has not happened, other users cannot edit these objects. See also chapter Editor / Closing the Editor. Closing the Editor.

If you close without accepting the changes, a dialog for accepting or refusing the changes is displayed:



Parameters	Description
List field elements	Lists all elements which are disabled for editing. Changes for the displayed elements can be accepted or discarded via the respective button.
Accept changes	The changes of all elements are accepted.
Undo changes	The changes of all elements are discarded.
Edit...	Closing is cancelled.
Next	Continues the closing process for the current project. The elements remain disabled for the other users. If several Multi-user projects should be closed, this dialog is displayed for the project which should be closed next.
Next (all)	Continues the closing process for all Multi-user project in the workspace which should be closed. The elements remain disabled for the other users.
Help	Opens online help.

9. Working offline

While working offline there is no connection to the server, e.g. when the engineer takes the project with him on the laptop.

To switch a project offline, the entry Activate offline mode has to be selected in the context menu of the project. Projects having no connection to the database server will be activated offline automatically.



Information

Accept changes and Reject changes is not possible during the offline operation.

Only **Allow changes** is available in offline mode, otherwise no changes would be possible in offline operation. **Accept changes** can be activated in the online period.

9.1 Switch offline

There might be different circumstances when switching offline

9.1.1 Elements/modules are locked by another user

At the moment of switching offline another user edits the element/module. In this case no changes can be done offline.

9.1.2 Elements/Modules are checked out by the same user during offline switch

At the moment of switching offline the same user edits the element/module. In this case no online changes by another user are possible.

9.1.3 No elements/modules are checked out during going offline

At the time of switching offline no elements/modules are checked out.

CASE 1:

Offline changes for an element/module are allowed, but the same is also possible online.

User A changes an element/module offline, user B online.

When user A goes online, it is recognized that user B is just editing the module/element online.

The changes made by user A are lost.

 **Attention**

In this case the changes made offline (by user A) are lost!

CASE 2:

When user A accepts his changes, user B already accepted his changes.

The changes made by user A overwrite the changes made by user B when going online.

 **Attention**

In this case the changes made online (by user B) are overwritten!

9.2 From offline to online

To switch an Offline-project online again, the entry Project/online has to be selected in the context menu. The project is reloaded into the Editor immediately.

9.3 Possibilities in offline operation

Elements/modules	Create, change, rename, delete
------------------	--------------------------------