Ribbon Menu

We have attempted to keep the content of the document complete, accurate and under permanent review.
However, due to the continuous development of described software, it is not possible to guarantee up to date and accurate information, integrity or quality of the content at all times.
As far as we are able to detect errors or omissions or those, which have been reported to us, our attempt is to correct these with the following versions. imos is not responsible for any direct or indirect damages caused by the use or non-use of the presented information or through the use of faulty and incomplete information. The descriptions in this document can be changed at any time without prior notice.

Creation date: June 2015
Used version: 10.0SR2 and AutoCAD 2013

Content

1. Definition 3
1.1 Customization file 3
1.2 Ribbon 3
1.3 Register, groups and commands 3
1.4 Enterprise-customization (CUIx) 4
2. Create New Menu 5
2.1 Copy and insert in the customization dialog 5
2.2 Create new CUI 8
2.3 Allocating available commands 14
2.4 Load partial customization file 15
2.5 Allocate specific project commands 18
2.6 Provide commands of several systems 21

1. Definition

1.1 Customization file

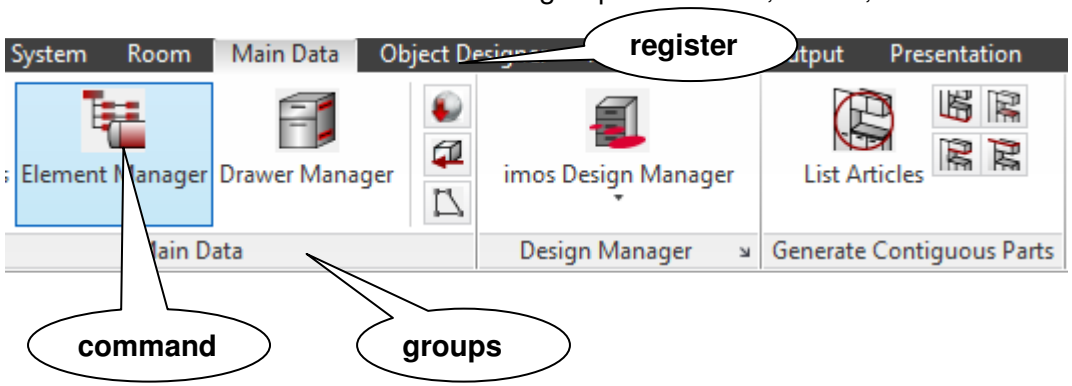
The previous applied MNU- and MNS files from AutoCAD have been replaced by a file type, the XML-based CUIx-file which is the successor of the CUI-file format implemented in AutoCAD 2006.
The XML-file defines most elements of the user interface.
The main -CUIx-file *imos.cuix* is automatically loaded when starting.
Partial CUIx-files can be loaded or deleted in a drawing session if required.

1.2 Ribbon

The Ribbon menu is based on a graphical application concept to combine control elements and icon-based toolbars. Command groups are formed and corresponding bundled commands are clearly shown when clicking tabs.

1.3 Register, groups and commands

The tab in the ribbon can contain several groups and these, in turn, several commands again.



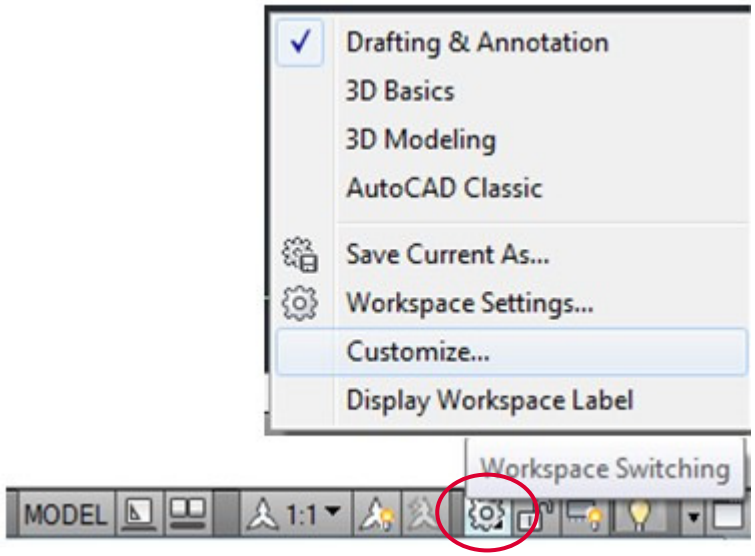
1.4 Enterprise-customization (CUIx)

Customization information is usually saved in an enterprise customization file shared by many users, but controlled by a CAD-Manager. Enterprise-CUIx-files facilitate the administration and edit customization files.
For preventing any unintentional modifications, this file is write-protected loaded and therefore not configurable anymore. However, if further commands have to be implemented, a partial customization file must be created.
This procedure is described below.

2. Create New Menu

2.1 Copy and insert in the customization dialog

For customizing the menu, the function customize must be called above the symbol **changing workspace** .

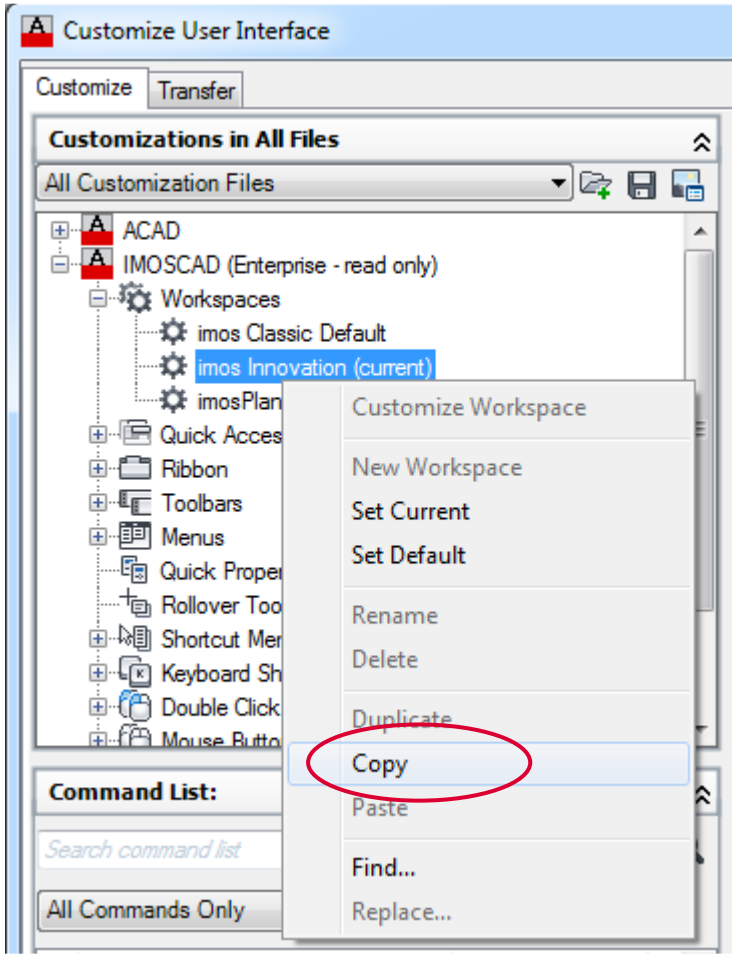


For implementing a new register in an existing workspace, this workspace must be copied and saved under a new name.

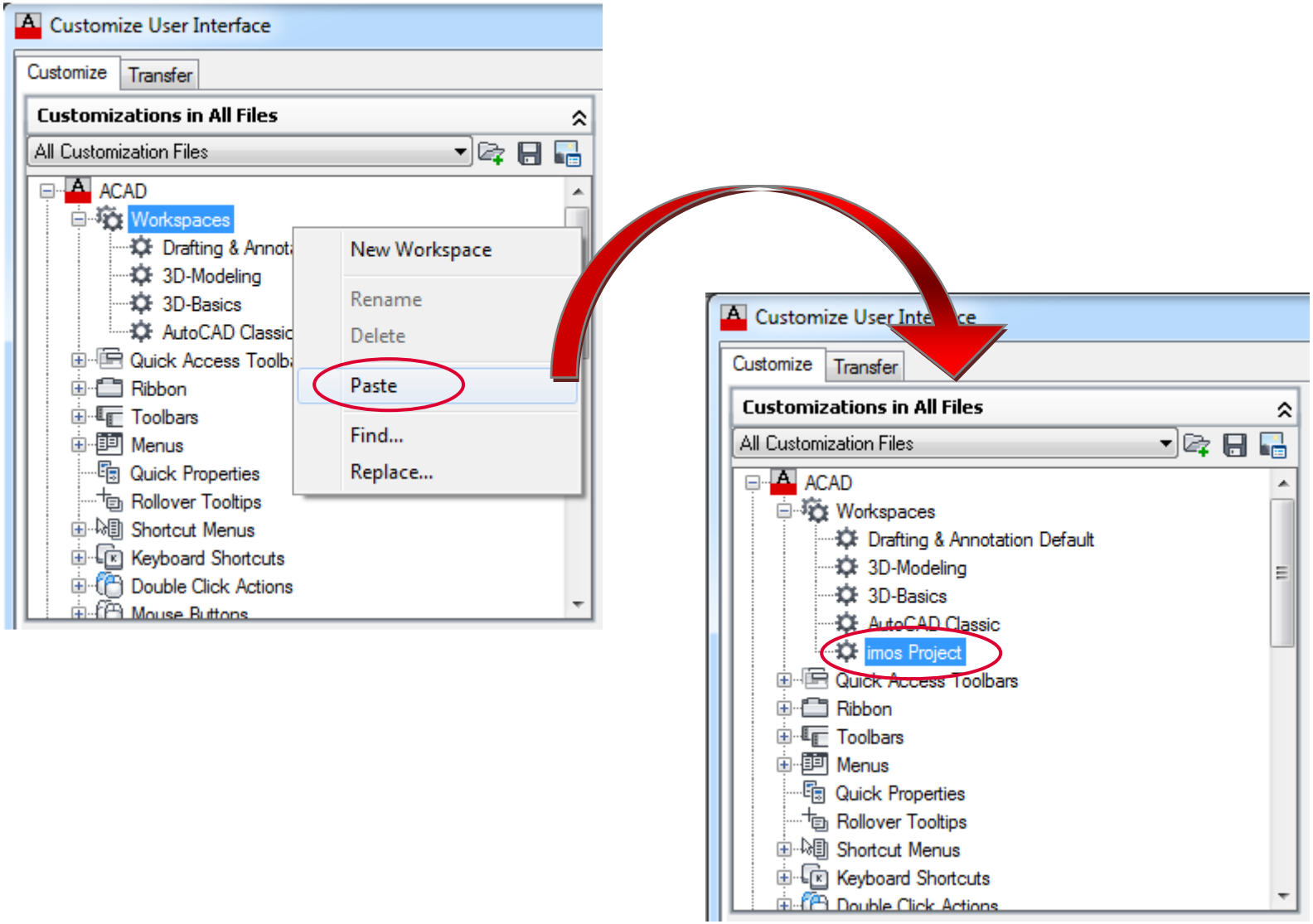


The main file **imos Innovation** is write protected and therefore cannot be changed!

In order to do this, the **workspace** for copying is marked by right clicking in the dialog **customize user interface**.

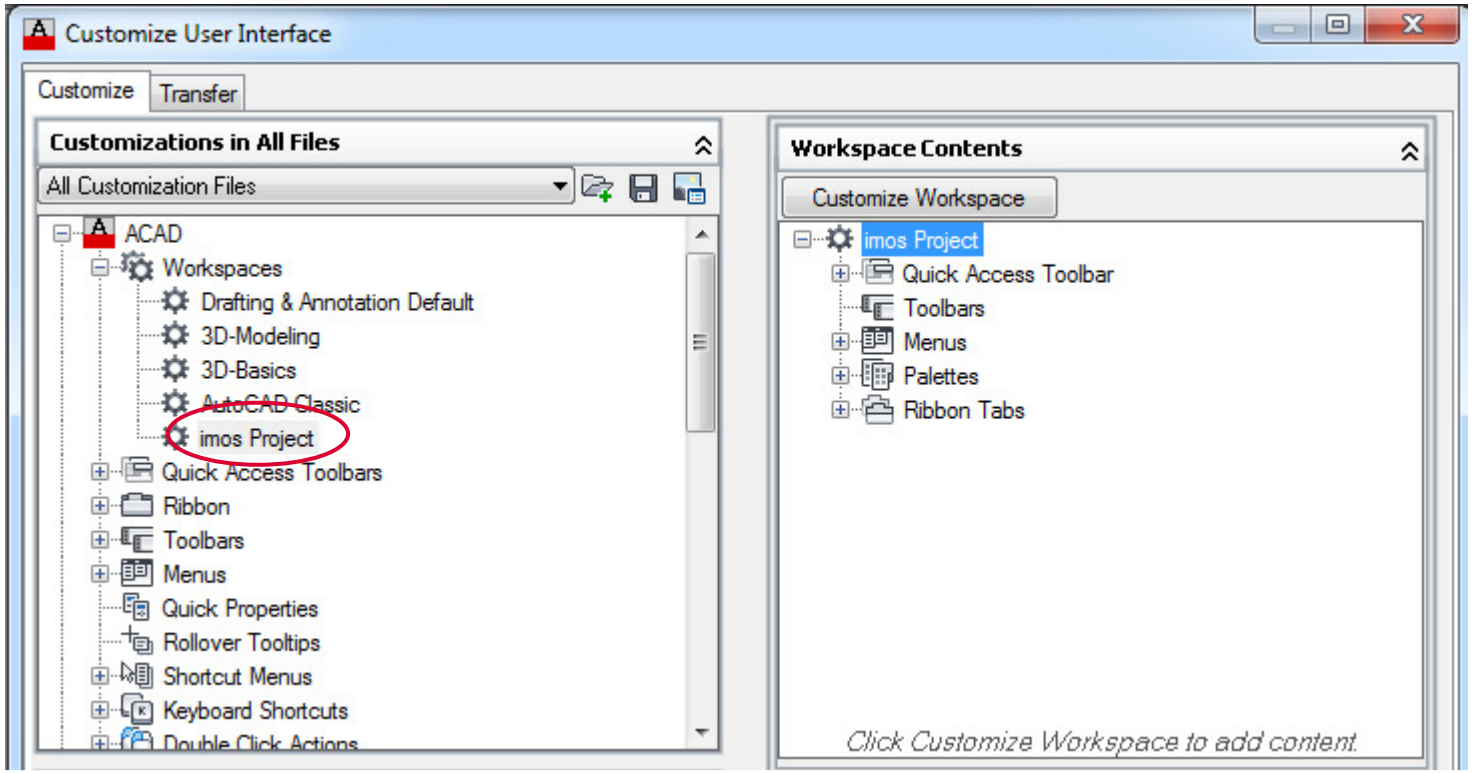


Afterwards this copied workspace is inserted in the not write protected space (in this case ACAD). In order to do this, the **workspace** must be marked by right clicking and the function **insert** called.



For avoiding any possible confusion, the workspace should be renamed via the context menu.

On the right side of the dialog the properties of the workspace are displayed. Currently, these properties are equivalent to the context of the **imos Innovation** menu.
All available tabs with allocated groups are displayed below the ribbon.



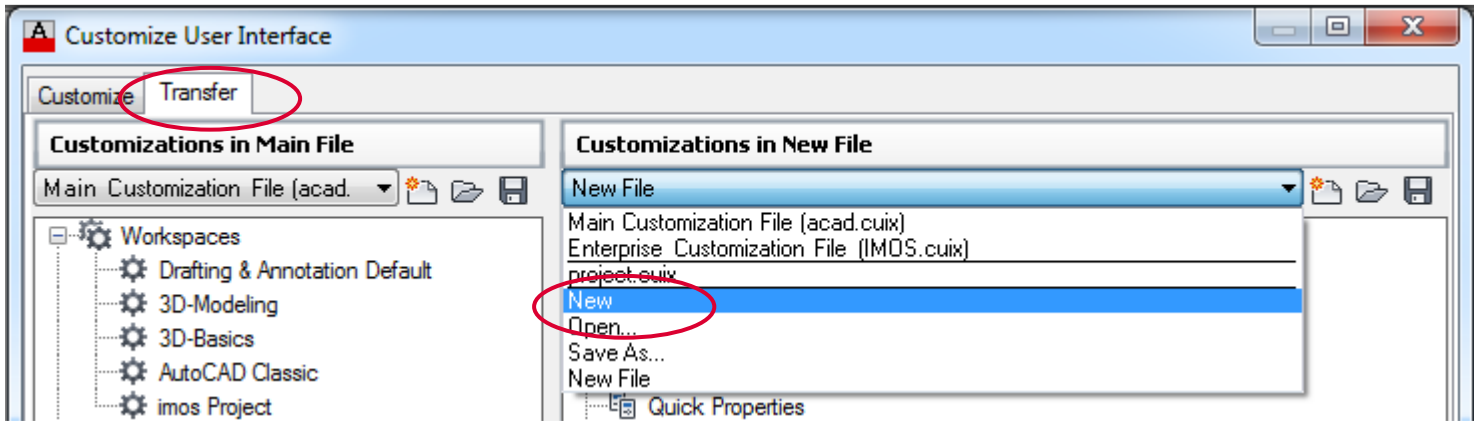
2.2 Create new CUI

After creating the workspace, new registers and groups can be applied now.

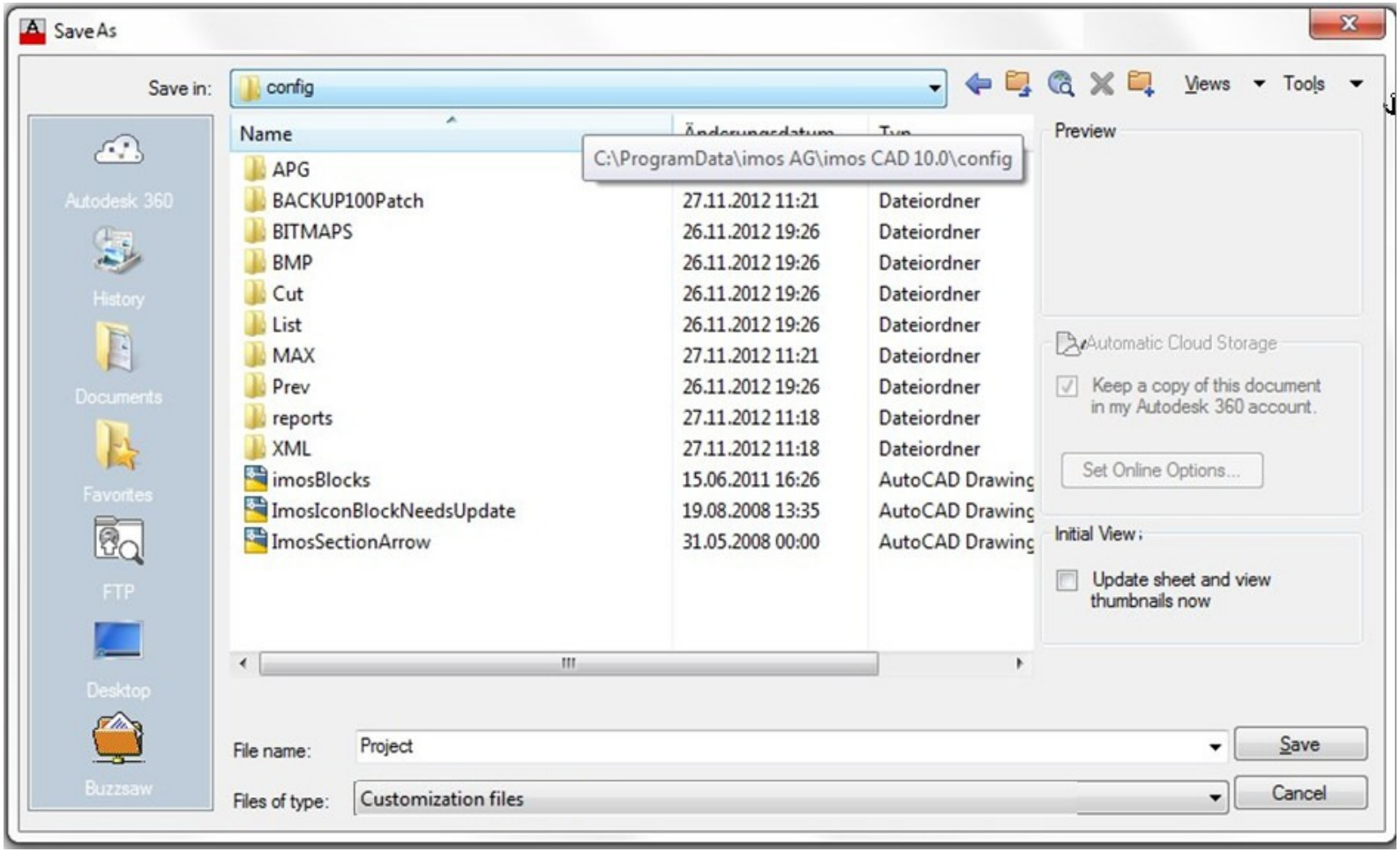
Ribbon Menü

For applying these on other workspaces later, they are saved as own CUI and therefore can be loaded partially afterwards.

In order to do this, the register must be changed to **transfer** and then a new customization file can be created there by calling the function **new** via the pull-down menu on the right side.

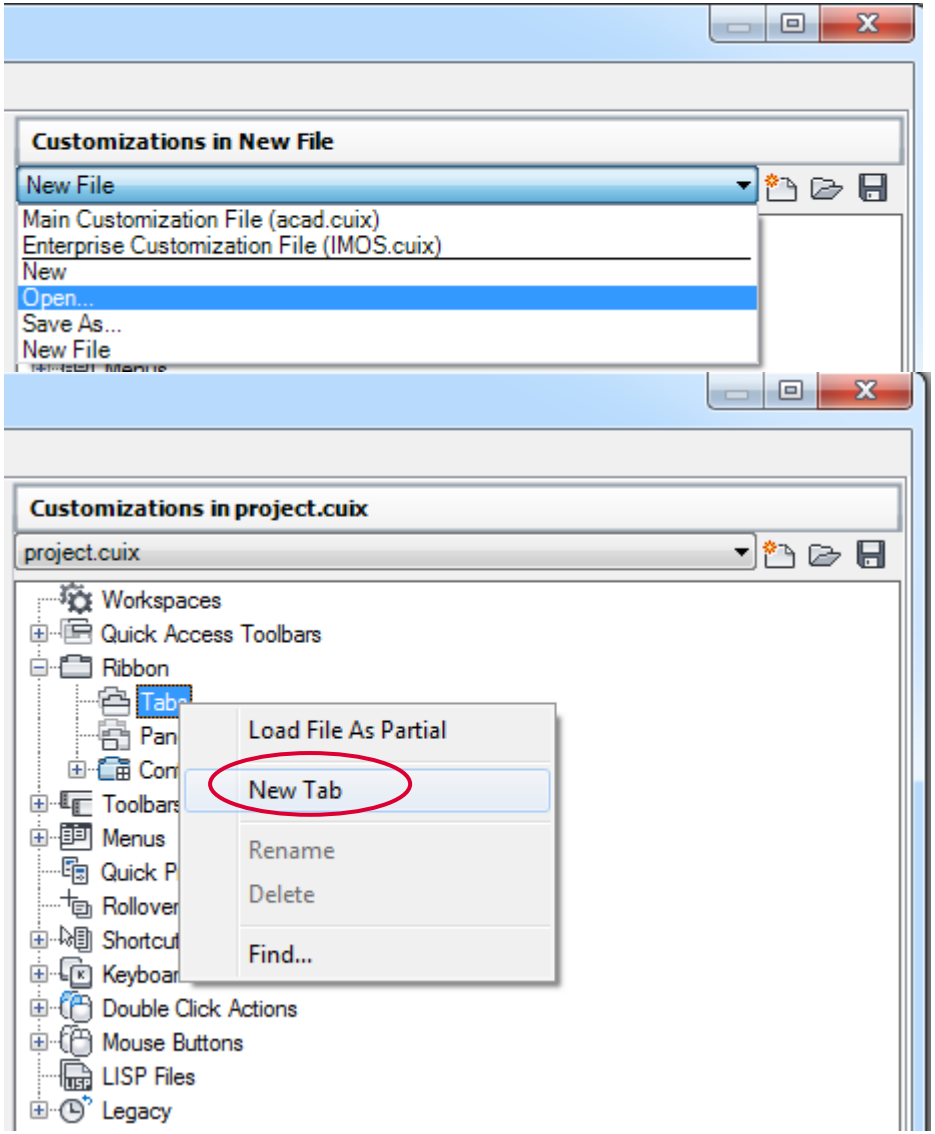


After that a new dialog opens to define the storage of the new customization file.



The file is saved in the **config** directory. In doing so, the name is freely defined!

In this example the file was named **Project.cuix.** in the following description this name is applied!



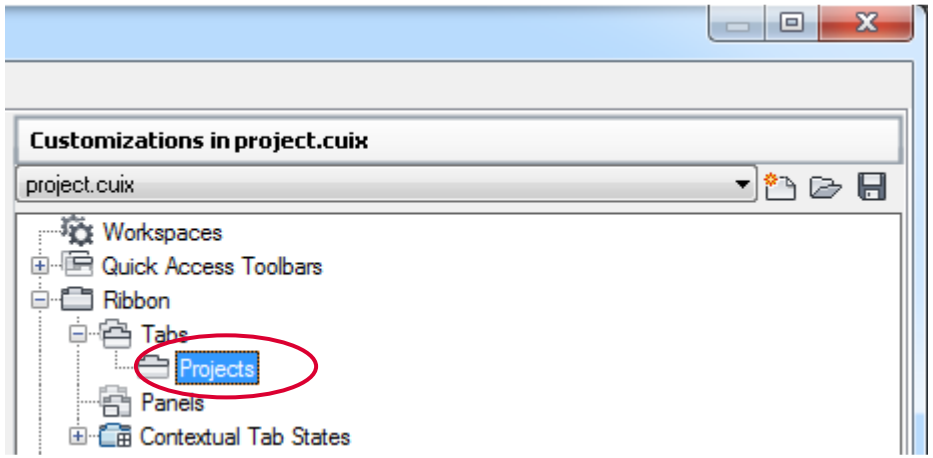
Currently, the new **Project.cuix** is still empty.
In order to fill it with content, the file must be loaded first. To do this, the function **open** is called via the pulldown menu and the new Pr**oject.cuix** is loaded on the **config** directory there.

First step is to apply a new tab for the new customization file in which the menus can be loaded later.

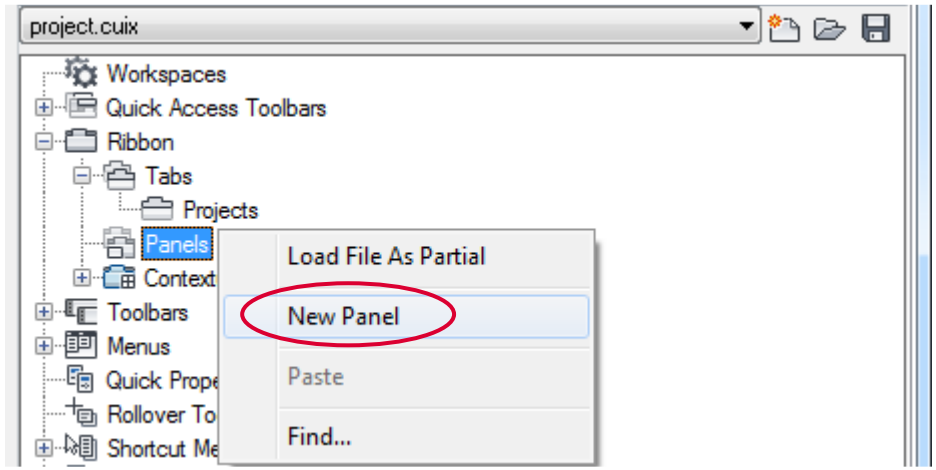
In order to do this, the entry tabs are marked by right clicking below the **ribbon** and the function **new tab** is called in the context menu.

The information about the current loaded CUIX is displayed on the right side of the dialog.

Thereby a new tab is applied below that still has to be renamed.

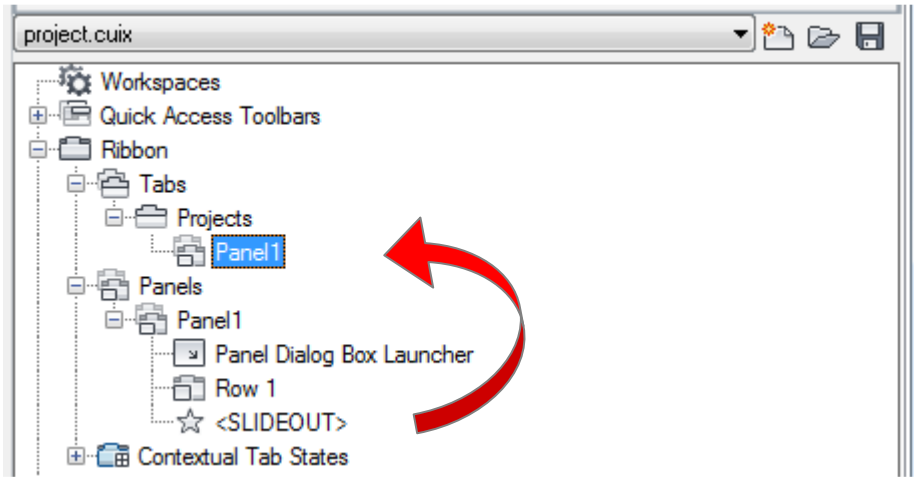


The second step is to create a new group which is allocated to the register later.
The procedure is the same as with the registers here. The function **new menu bar** is called via the context menu!



In the current status this register contains no groups and no commands yet!

Afterwards this new group is allocated to the new register **per Drag&Drop**.

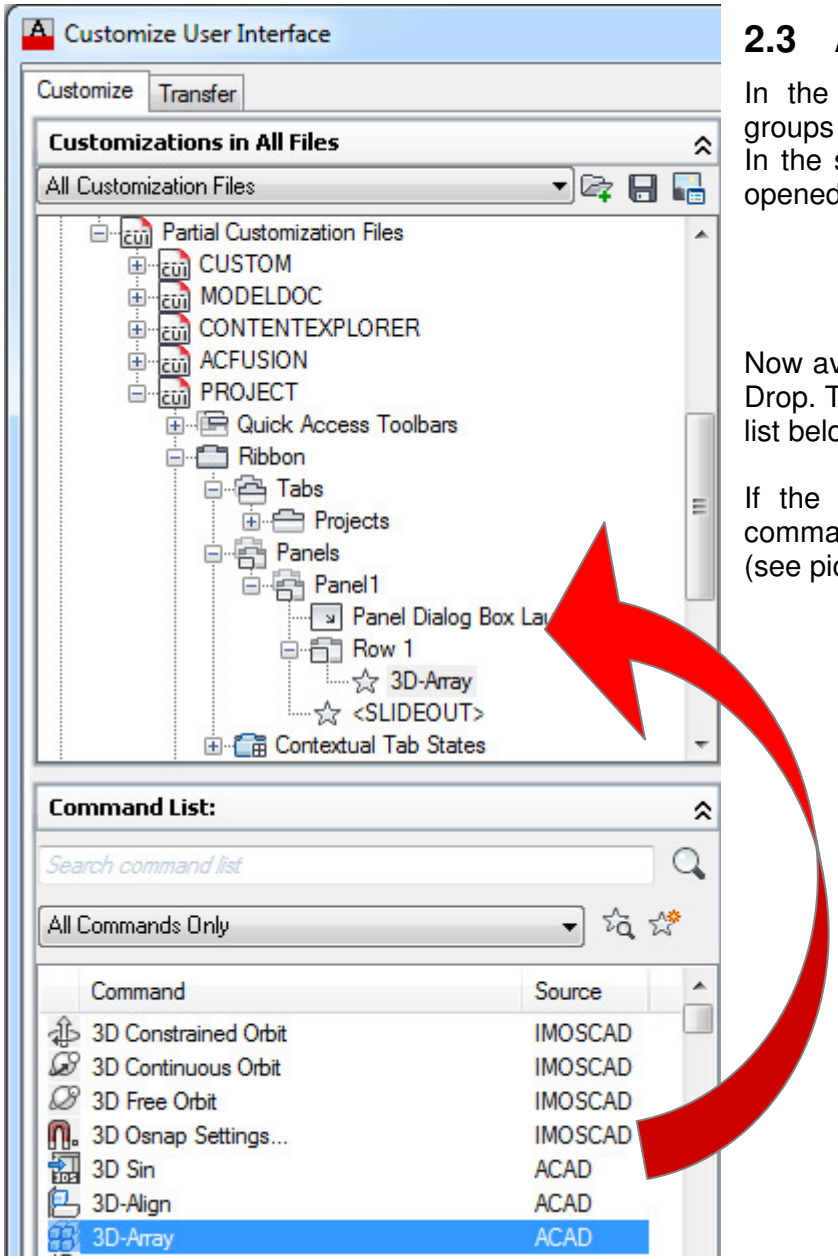


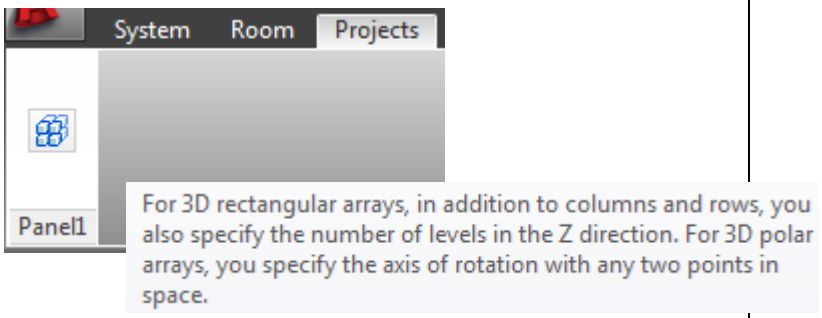
2.3 Allocating available commands

In the following step, commands should be allocated to the groups now.
In the subdirectory of the Project.cuix the entry **ribbon** must be opened.

Now available commands can be dragged into the file per Drag& Drop. To do this, the command must be selected in the command list below and then dragged to the group above.

If the customization file is partially loaded, the command or commands appear in the menu.
(see picture below)

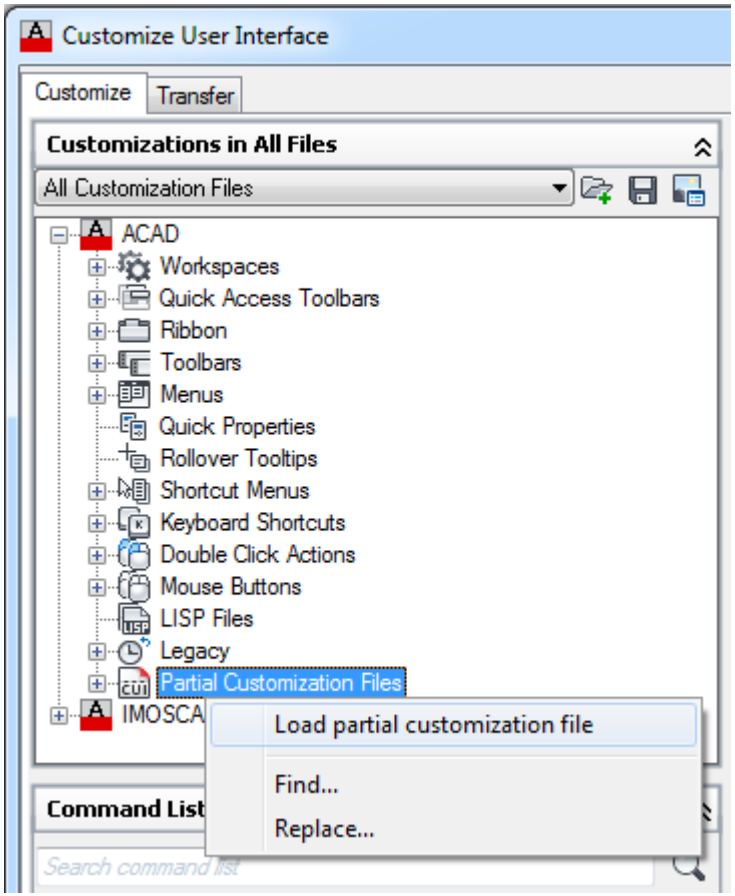




Ribbon Menü

2.4 Load partial customization file

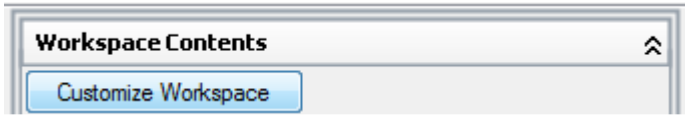
For loading the new created customization file partially, change to the register **customize** in the left part of the dialog. There the function **partial customization files** must be marked by right clicking under **customization files** and then the function **load partial customization files** has to be selected. ´



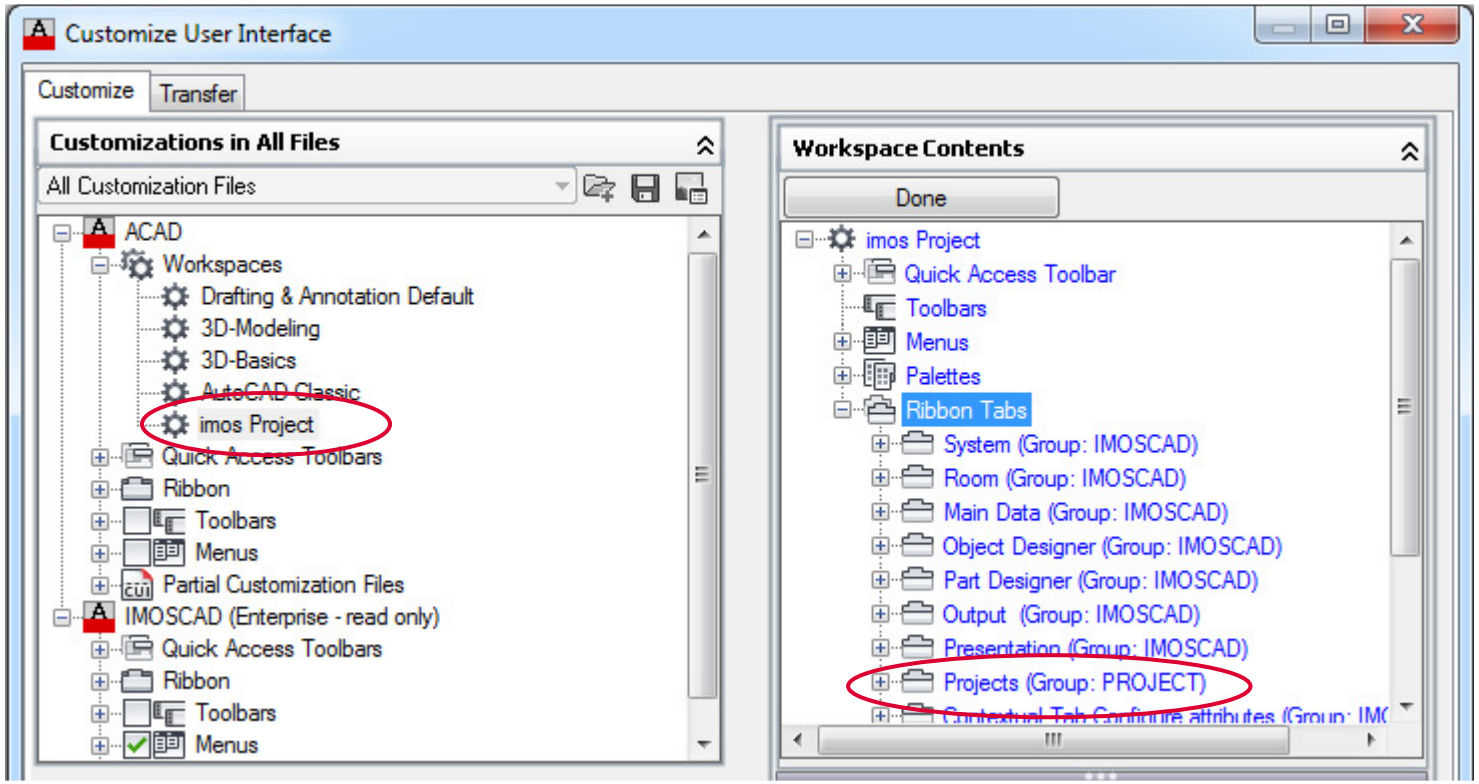
In the appearing dialog the applied **Project.cuix** from the **config** directory must be opened then.

Partial loading means that the new created register is only added to the already available registers but does not replace them.

For checking, if the file has been correctly loaded, the workspace **imos Project** can be customized. For this purpose, it is marked on the left side and the function **customize workspace** must be selected afterwards.

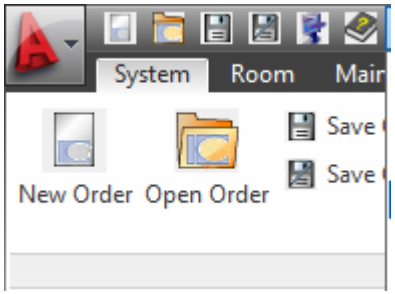


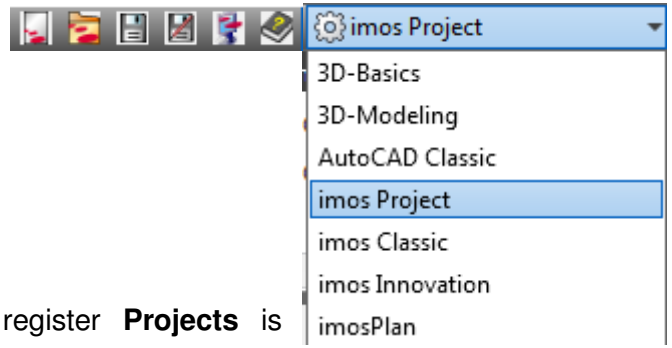
Then the CUIX appears on the right.



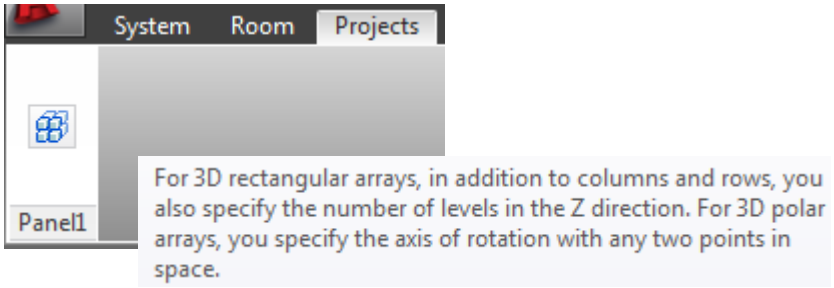
If the CUIX has been correctly loaded, then leave the dialog **customize user interface** via the **OK** Button on the right below.

Now the workspace **imos Project** can be selected from the menu.



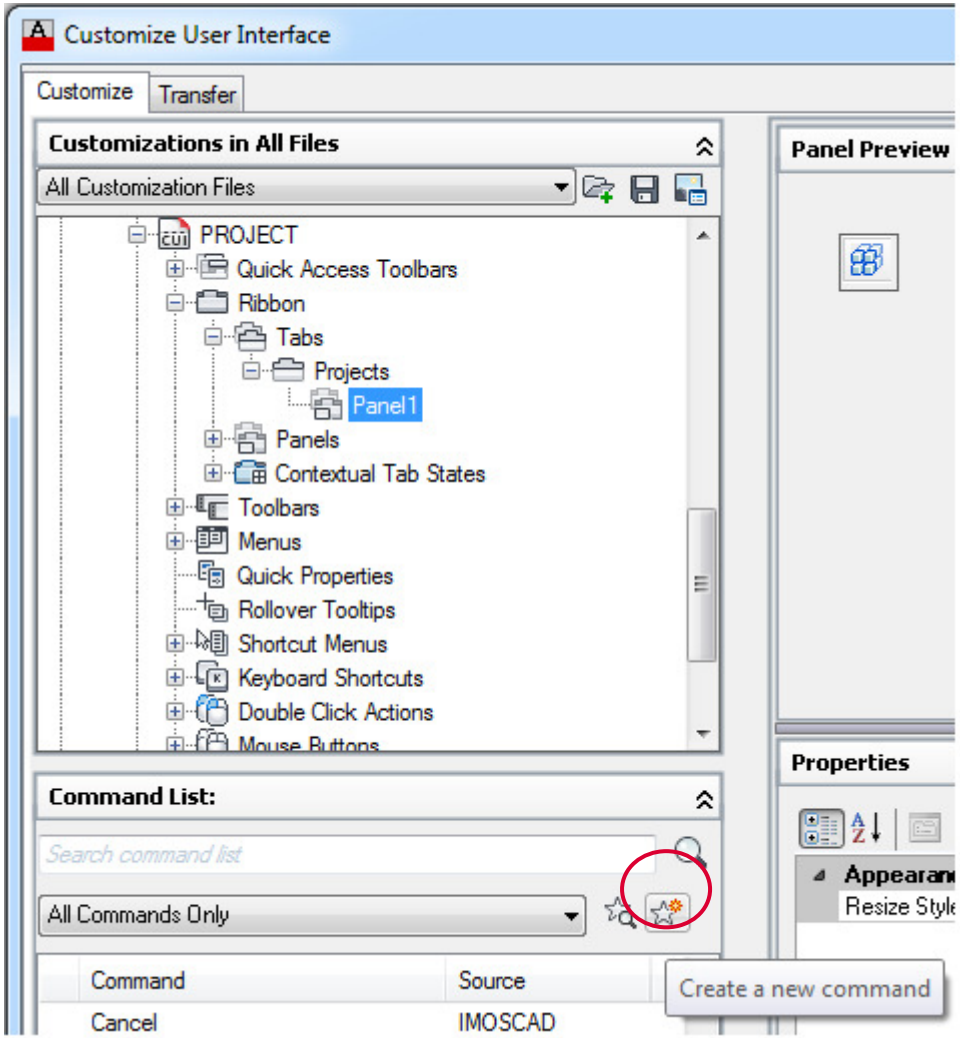


After loading the workspace, the register **Projects** is available as well!



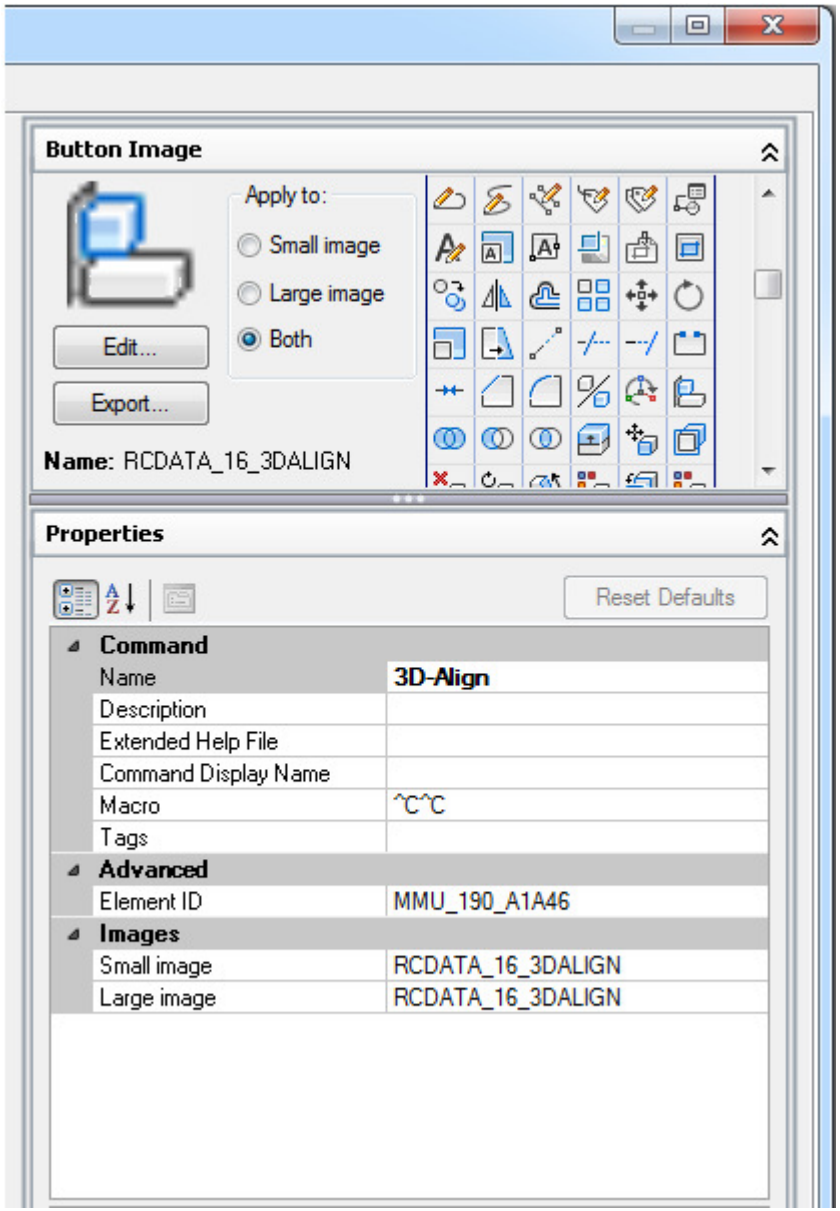
2.5 Allocate specific project commands

If a specific project command has to be applied in a group, that is not available in imos or AutoCAD in general, the procedure is described as follows.



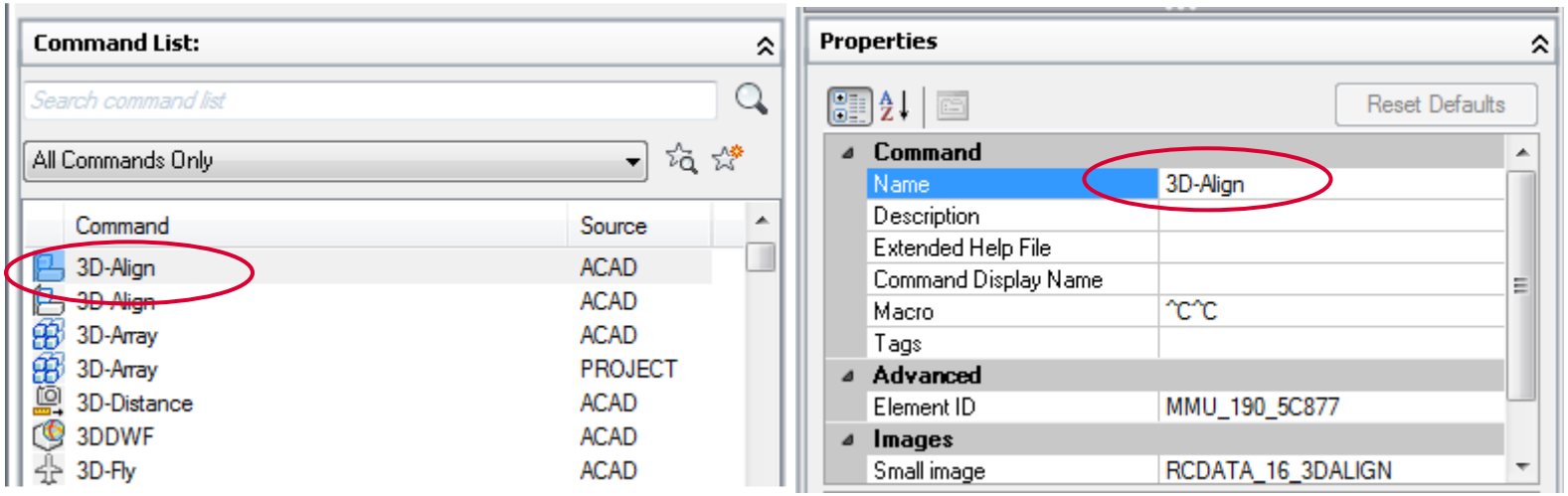
First the partial loaded **Project.cuix** is selected as before and the function **create command** is selected afterwards.

Ribbon Menü

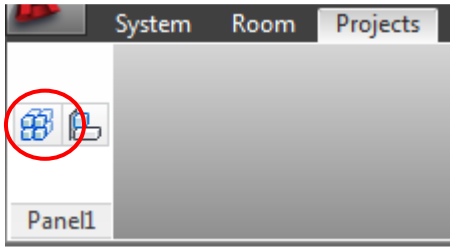


After that the display changes on the right side of the dialog and a new command can be defined there.

Then this command is also listed on the left side in the list of the commands and therefore can be dragged into the group per **Drag&Drop**.



By leaving the customization dialog via OK, the new command is adopted in the group.



2.6 Provide commands of several systems

The previous described procedure in **2.3** only involves providing loaded commands for the current system.
For providing the Project.cuix for other systems as well, then proceed as shown in point **2.4** Load partial customization file**.**

First the **project.cuix** must be selected from the pulldown menu above and then commands can be allocated per Drag&Drop as before.

