
pure::variants - Connector for Zuken CR-8000 Manual

Parametric Technology GmbH

Version 6.0.8.685 for pure::variants 6.0

Copyright © 2003-2024 Parametric Technology GmbH

2024

Table of Contents

1. Introduction	1
1.1. Software Requirements	1
1.2. Installation	1
1.3. About this manual	1
2. Using the Connector	2
2.1. Data Flow Overview	2
2.2. Using the pure::variants Integration for Zuken CR-8000	2
2.3. Importing Variation Information from Zuken Files	16
2.4. Defining New Destinations for the Zuken Design Model	19
2.5. Creating Zuken Destinations by Transformation	19
3. Known Restrictions	20

1. Introduction

The pure::variants Connector for Zuken CR-8000 enables users to formally manage the variability within Zuken CR-8000 Design Gateway projects by controlling Variation Type, Variation Item and Destination definitions within Zuken via tool integration to pure::variants.

1.1. Software Requirements

The following software has to be present on the user's machine in order to support the pure::variants - Connector for Zuken CR-8000:

Tool: Zuken CR-8000 Design Gateway 2018 is required. Compatibility with other Tool releases is not guaranteed.

The pure::variants Connector for Zuken CR-8000 is an extension for pure::variants and is available on all supported platforms.

1.2. Installation

Please consult section **pure::variants Connectors** in the **pure::variants Setup Guide** for detailed information on how to install the connector (menu **Help** -> **Help Contents** and then **pure::variants Setup Guide** -> **pure::variants Connectors**).

As a second step, the pure::variants specific tool integrations of CR-8000 need to be installed. Before proceeding with this step, make sure Zuken CR-8000 is installed on user's machine.

Please consult section **pure::variants Integrations** in the **pure::variants Setup Guide** for detailed information on how to install the connector (menu **Help** -> **Help Contents** and then **pure::variants Setup Guide** -> **pure::variants Integrations**).

1.3. About this manual

The reader is expected to have basic knowledge about and experiences with both tools, Zuken CR-8000 Design Gateway and pure::variants, and must be aware of the processes how define and manage Destinations in CR-8000, including Variation Types and Variation Items.

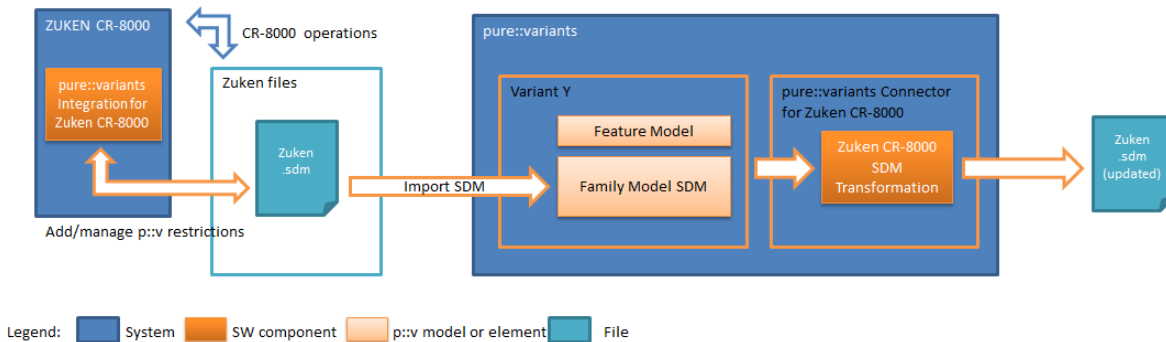
The pure::variants manual is available in online help as well as in printable PDF format [here](#).

2. Using the Connector

2.1. Data Flow Overview

Following figure describes the high level data flow, the imports and transformation outputs (Figure 1, “High level data flow”).

Figure 1. High level data flow



The connector described in this manual covers the transformation 'pure::variants Connector for Zuken CR-8000'.

2.2. Using the pure::variants Integration for Zuken CR-8000

To support users of the pure::variants Connector for Zuken CR-8000, the pure::variants integration provides an editor for variability information (see [the section called “Adding and Editing Variability Information”](#)) and visualizations to preview variants and find errors in variability information (see [the section called “Finding Problems in the Variations”](#)), directly in Zuken CR-8000.

First Use of the Integration


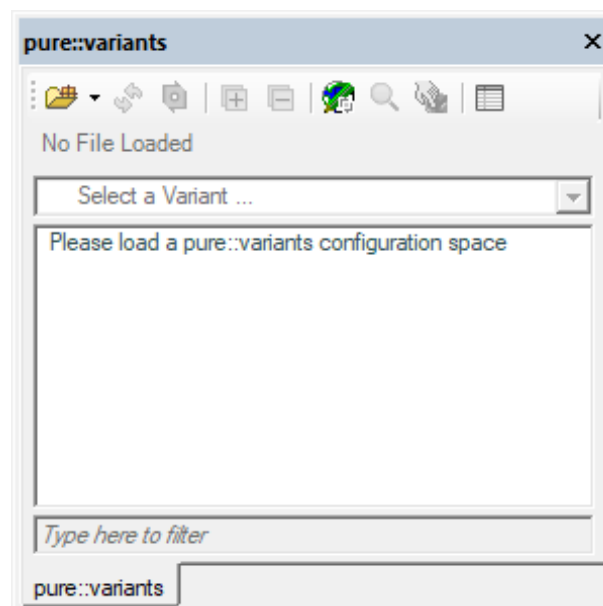

To open the Integration window, first open a sheet, and then click on  (pure::variants Integration for Zuken) in the toolbar. Now the Integration window as shown in [Figure 2, “pure::variants Integration Window”](#) should be visible.

Figure 2. pure::variants Integration Window



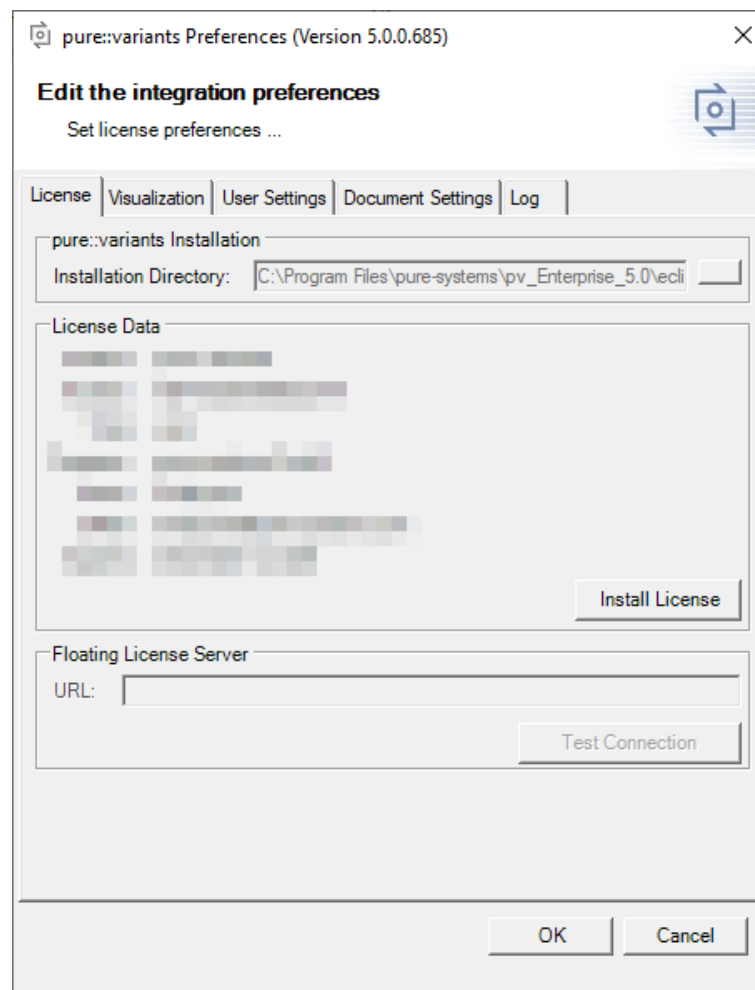
When you first use the Integration after installation, it is necessary to check whether the license preferences are correct. To this end, open the Integration preferences dialog via the  button in the Integration window.

A dialog opens that shows the path to your pure::variants installation and your license information (see [Figure 3, “Preferences Dialog”](#)). If any of the information is missing, you need to enter it. Use the ... button in the **pure::variants Installation** group to enter the installation directory, and the **Install License** button to specify your license.

If you are using a floating license and the URL in the **Floating License Server** group is not set already, you need to enter the URL. To test if the connection to the floating license server is established, press the button **Test Connection**.

Now you can use the Integration.

Figure 3. Preferences Dialog



Connecting with pure::variants Models

For editing variability information and viewing visualizations, it is necessary to connect your Zuken CR-8000 project with one or more pure::variants models. The following types of pure::variants models can be loaded:

- Recommended: pure::variants configuration spaces, which enable selection of contained variant description models (.vdm)
- pure::variants variant result models (.vrm)
- pure::variants feature models (.xfm)

- pure::variants family models (.ccfm)

pure::variants models can be opened from two different sources: Either from a *pure::variants/Eclipse workspace* or from a *pure::variants model server*.

Opening models from a workspace



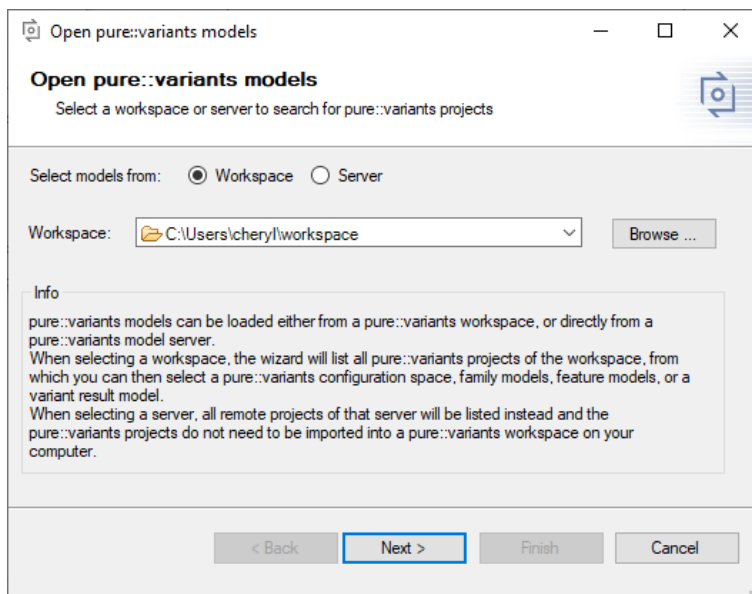
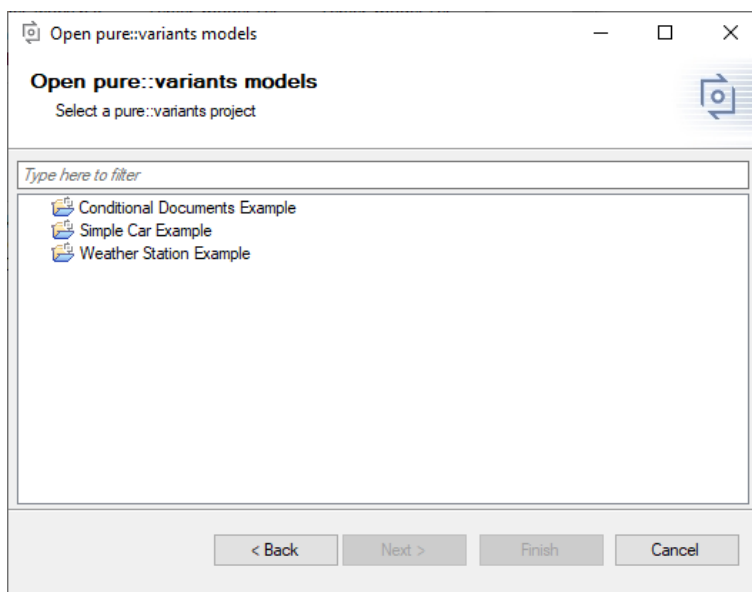
To open a model or configuration space, press  on the Integration window. This will open a wizard, which first allows choosing the source (workspace or server). Choose *workspace* and then browse to find your pure::variants workspace folder. Already known workspaces are listed in the workspace dropdown box. If you later need to add or remove a workspace from the list, you can go to tab *User Settings* of the Integration preferences (accessible via .

Figure 4. Mode selection page



On the next page, all projects are listed that are located in the selected workspace folder or that are linked into the pure::variants/Eclipse workspace.

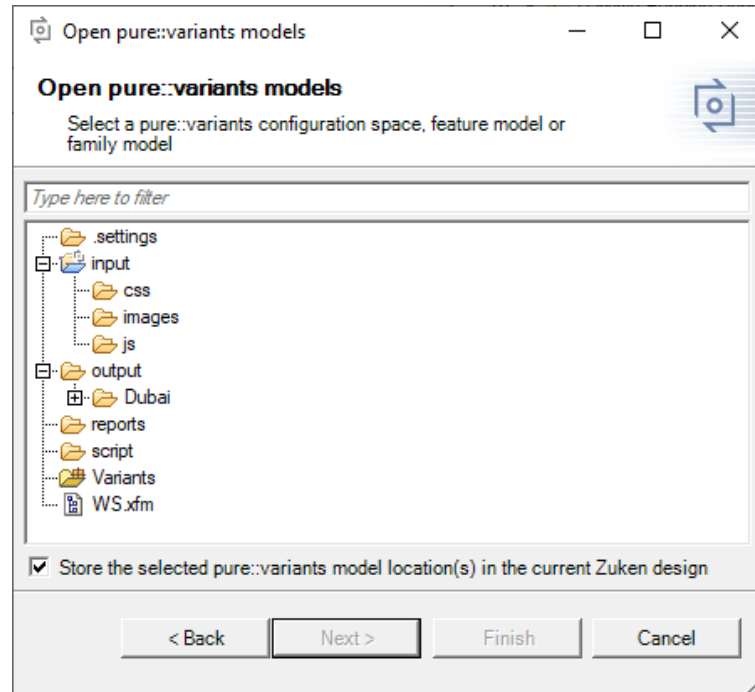
Figure 5. Project selection page



Select one and on the next page choose the model(s) or configuration space you want to open.

Checkbox *Store the selected pure::variants model location(s) in the current Zuken design* allows you to save the selected model locations in the current Zuken design, so that these models will be opened again once any user opens this Zuken design. If you do not select the checkbox, the model locations will only be stored on your computer. For details, see [the section called “Saving and Loading pure::variants Models”](#).

Figure 6. Model selection page



Opening models from a model server



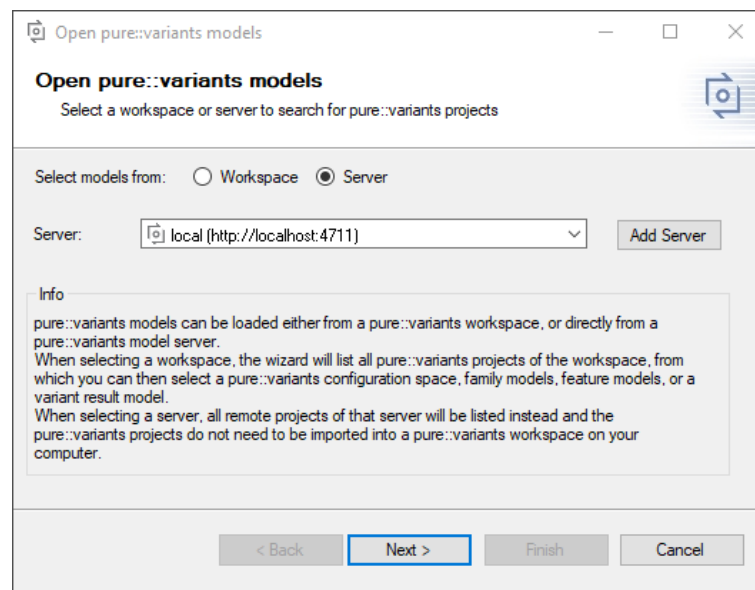
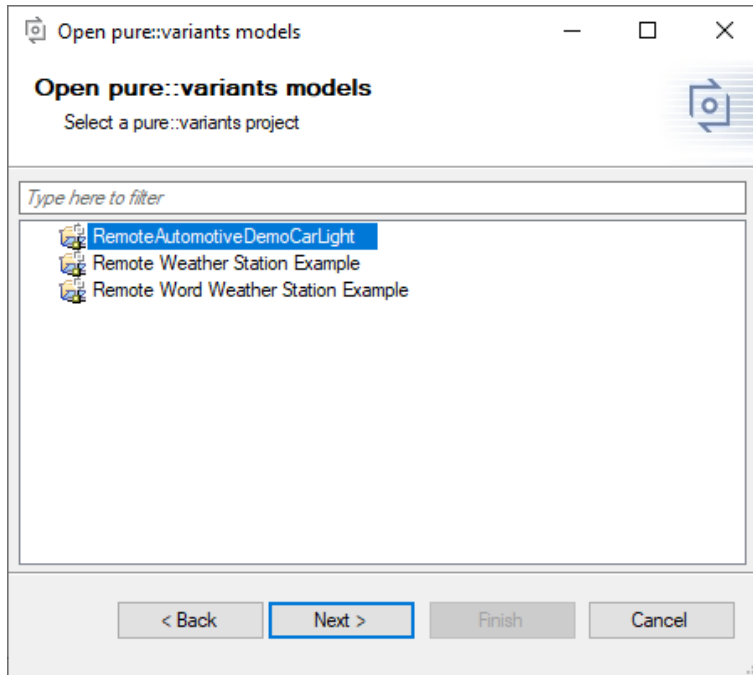
To open a model or configuration space directly from a pure::variants model server, also press . On the first page of the wizard, choose *server* and add the server address via button **Add Server**. Like in pure::variants, new servers need a name and the server address (e.g., `https://yourserveraddress:443`). Any known servers are listed in the server dropdown box. If you later need to add or remove a server from the list, open the Integration preferences by pressing . On tab *User Settings*, you can add or remove servers.

Figure 7. Mode selection page



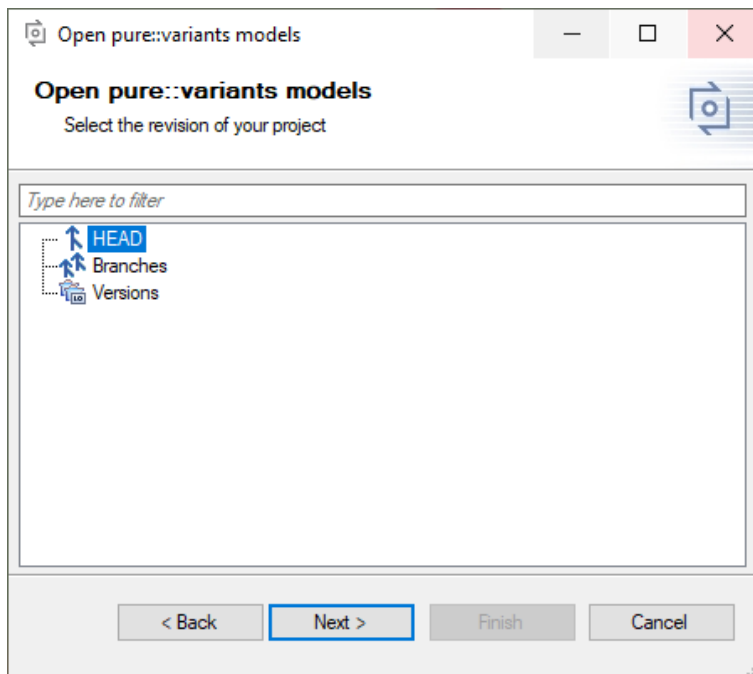
On the next page of the wizard, all projects of the server that the current user has read access to are listed.

Figure 8. Project selection page



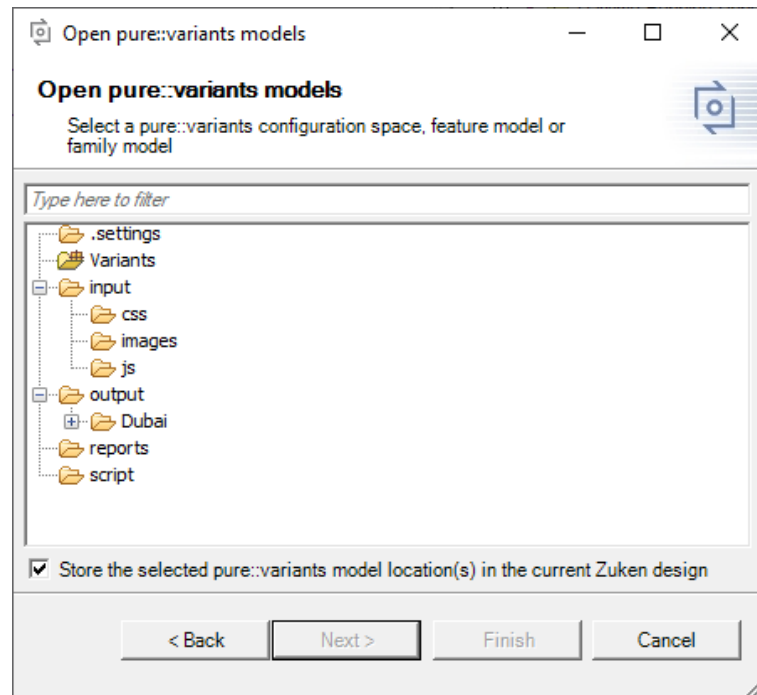
Select one and on the next page choose the project's revision (branch or tag) from which you want to load a model.

Figure 9. Project selection page



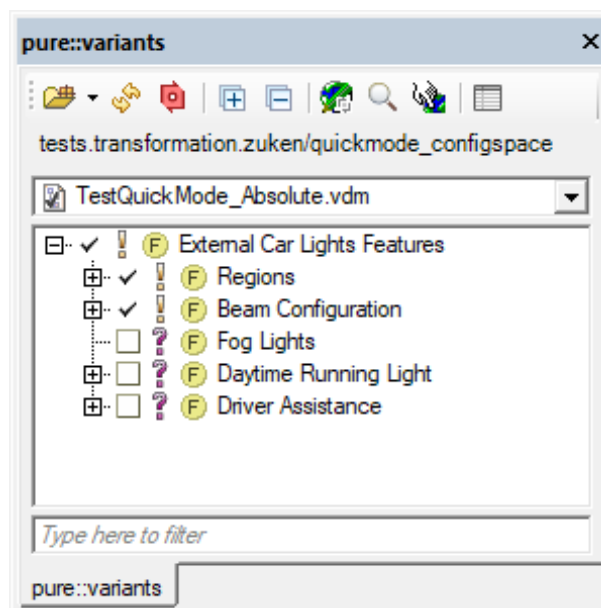
Finally, on the last page select the model(s) or configuration space you want to open.

Checkbox *Store the selected pure::variants model location(s) in the current Zuken design* allows you to save the selected model locations in the current Zuken design, so that these models will be opened again once any user opens this Zuken design. If you do not select the checkbox, the model locations will only be stored on your computer. For details, see [the section called "Saving and Loading pure::variants Models"](#).

Figure 10. Model selection page


Opening pure::variants Configuration Spaces

To open a pure::variants configuration space, use the wizard as described above. On the last page, select a configuration space folder. Now the Integration window should show all used models of your configuration space. Please note that family models (.ccfm) are not opened per default. You can enable loading family models in the Integration preferences on the **Visualization** tab. After selecting a variant from the dropdown list, selections should be shown in front of features. To ease usage of configuration spaces with many variants, the latest opened variants are shown at the top of the list.

Figure 11. Configuration Space with Selected Variant

¹You can create a variant result model in pure::variants by clicking the **Save Result to File** button that is shown in the toolbar of a variant description model.

Opening Other pure::variants Models

Other pure::variants models, such as variant result models¹ (.vrm), feature models (.xfm), and family models (.ccfm) can also be opened via . Please note that family models (.ccfm) are not listed per default. You can enable loading family models in the Integration preferences on the **Visualization** tab.


Live Connection with pure::variants

Since pure::variants 4.x, changes of the loaded pure::variants models are propagated live to the Integration. For example, directly after editing the name or changing the selection of a feature the loaded models are updated in the Integration window. To enable this live update, the following prerequisites need to be fulfilled:

- the opened model needs to be located in an Eclipse workspace
- the changes have to be done on the same Eclipse workspace using pure::variants 4.x or later
- either a configuration space, feature model or family model needs to be loaded (Variant result models can only be updated automatically when the .vrm file is saved)

Saving and Loading pure::variants Models

To ease the work with pure::variants Integrations, the last loaded model locations are saved, so that the model will be opened again automatically, next time you start the tool. Per default, these model locations are saved only for you on your local machine. If you want to save the last loaded model locations for all users, who are using a certain Zuken design, you can select checkbox *Store the selected pure::variants model location(s) in the current Zuken design* on the last page of the open model wizard. Then everytime a user opens that Zuken design, the pure::variants models stored in the document will be opened instead of the locally stored models (if they can be found on the user's machine).

Furthermore, a list of the latest loaded models can be accessed via the small arrow next to the  button.

Please note that models are saved relative to your current workspace, which is the workspace where your current model is located in or linked to. Therefore, you may be asked for your current workspace location when loading a model from a different workspace, or a model that is not located in a workspace (but may be linked into a workspace). Hint: If you want to know where exactly the loaded model is located, you can hover over the name of the model. A tooltip will show the full path of your currently loaded configuration space or model.

If you do not want to load a model again on startup, you can clear the stored model locations from the user and/or document settings. For details, see [the section called “Manage Settings”](#).

Model Visualization Preferences


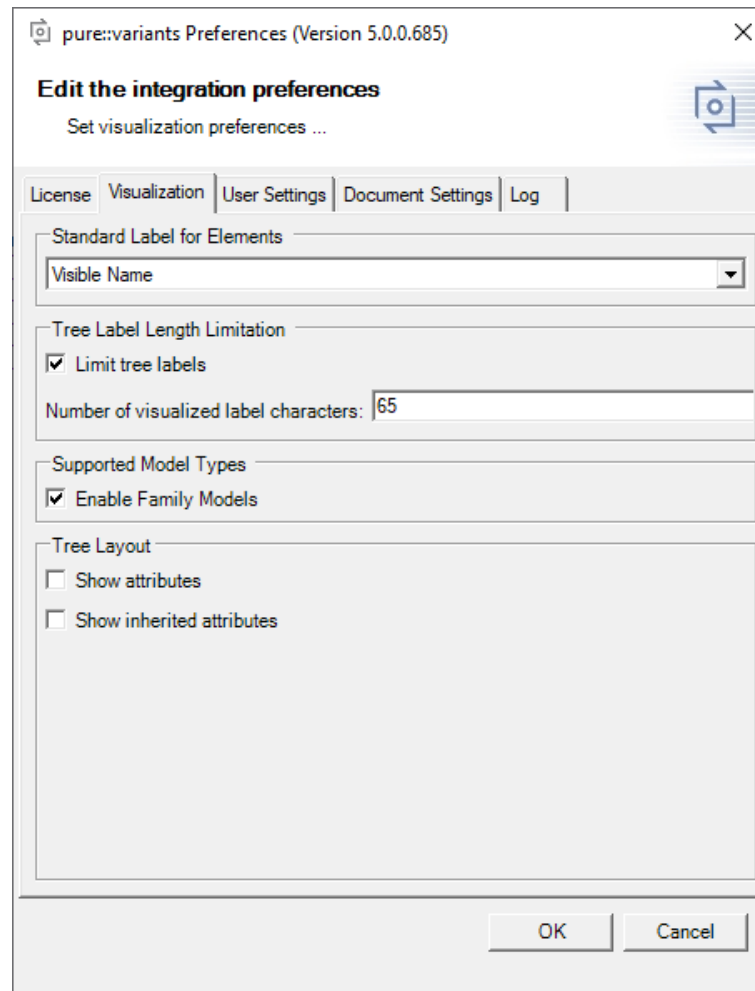

In the Integration preferences, you can set how pure::variants models will be displayed and which model types are supported. To do that, open the Integration preferences by pressing  and go to the **Visualization** tab (see [Figure 12, “Preferences Dialog Visualization Tab”](#)). The first dropdown box enables you to set how elements in the pure::variants model view are labeled. Furthermore, you can limit how many characters are shown for each element in the tree, enable or disable the loading of family models, and set whether attributes are shown in the model tree. To also show attributes inherited from parent elements, select **Show inherited attributes**.

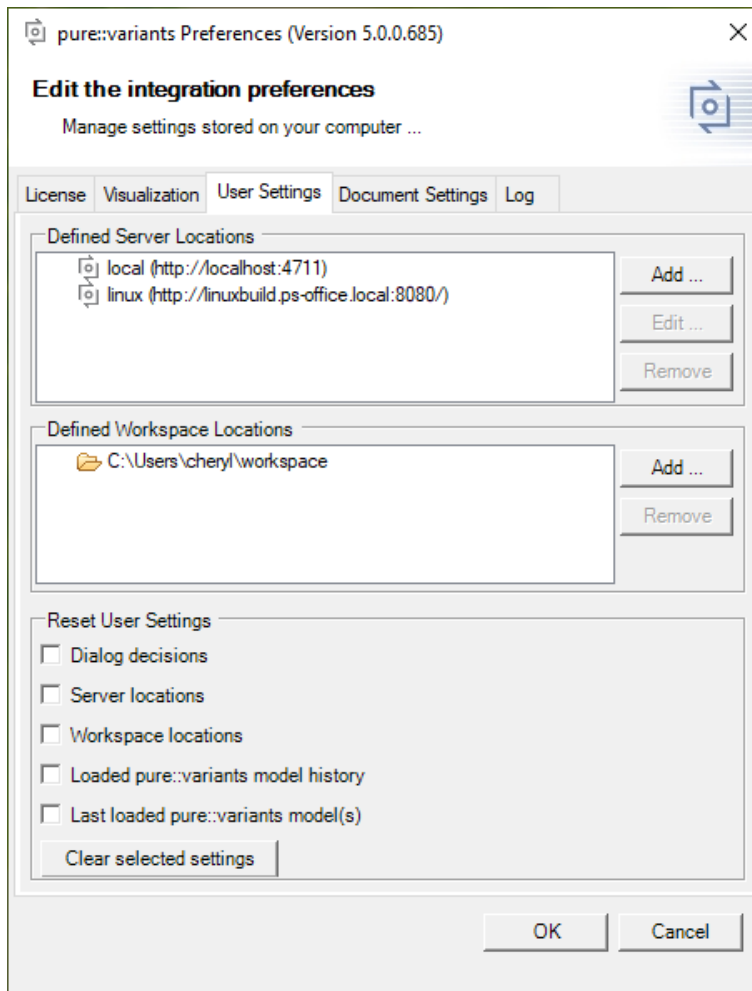
Figure 12. Preferences Dialog Visualization Tab

Manage Settings

In the Integration preferences, you can also manage settings stored by the Integration. You can, for example, add or remove known workspace and server locations or clear certain settings. To do that, open the Integration preferences by pressing .

Here, you can find tabs *User Settings* and *Document Settings*.

On tab *User Settings*, you can manage settings that are stored only on your local machine, specifically for your user. (see [Figure 13, “Preferences Dialog User Settings Tab”](#)). This includes server and workspace locations, dialog decisions, the history of previously loaded pure::variants models, and so on. The first section, **Defined Server Locations** enables you to add, edit and remove the server locations that are not locked. The second section, **Defined Workspace Locations** enables you to add and remove the workspace locations. The last section **Reset User Settings** enables you to clear the selected settings. For example, you can select checkbox *Last loaded pure::variants model(s)* and press the clear button, to make sure no pure::variants model is loaded from the user settings at startup of the Integration.

Figure 13. Preferences Dialog User Settings Tab

On tab *Document Settings* you can manage all settings that are stored in the current Zuken design. For example, you can clear the last loaded pure::variants model from the current document, so that, when you or other users next open the current Zuken design, the Integration will load no pure::variants model stored in the document.

Adding and Editing Variability Information


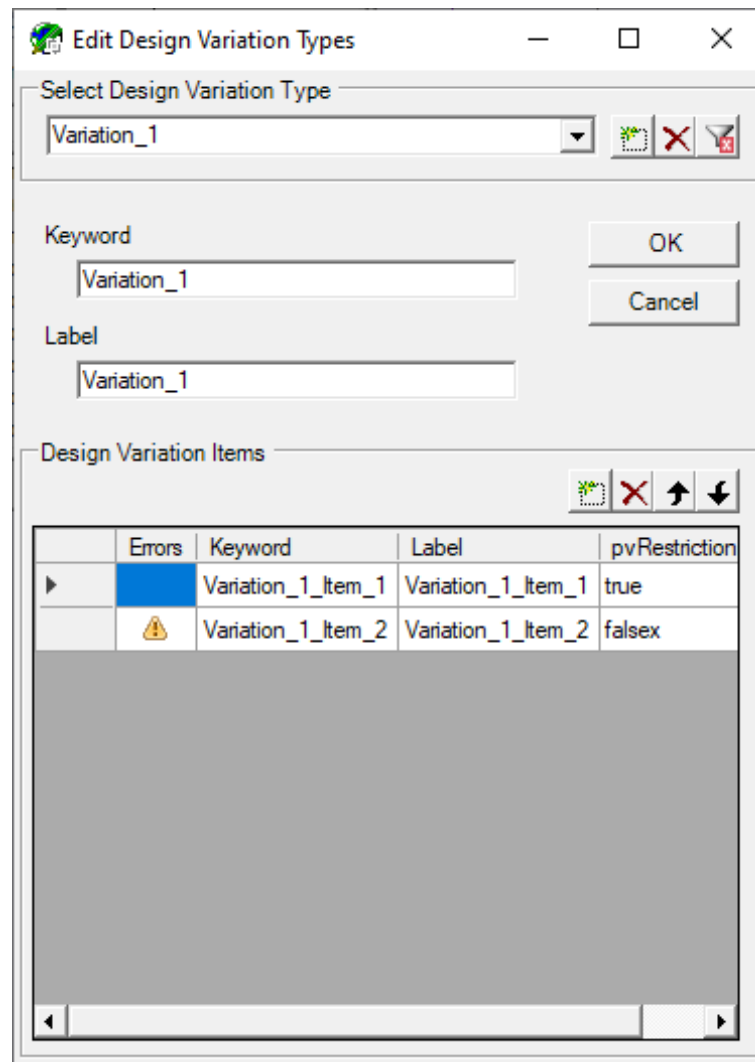






The variation types and variation items can be edited by using the "Edit Design Variation Dialog". You can open it by pressing  on the Integration window. Now the Integration window as shown in [Figure 14, "Edit Design Variation Dialog"](#) should be visible. This dialog can be used to add/edit/remove the variation types and variation items. You can see the different variation types by clicking on the dropdown box in the top of this dialog. For every variation type, its keyword, label and the different variation items are also visible.

Figure 14. Edit Design Variation Dialog

Adding, Editing and Deleting a Variation Type

To add a new variation type, click  under the "Select Design Variation Type" group. This creates a new variation type with two variation items in it. The Keyword and the Label of this variation type can be also edited in this dialog. To delete a variation type, click  button under the "Select Design Variation Type" group.

Adding, Editing and Deleting a Variation Item

The variations items can be added to the currently selected variation type by clicking  under the "Design Variation Items" group. This creates a new item with a default keyword and label. The items can be edited by clicking on the grid. To delete a variation type, click  under the "Design Variation Items" group. The index of the variation items can also be changed by using the  (move up) and  (move down) buttons.

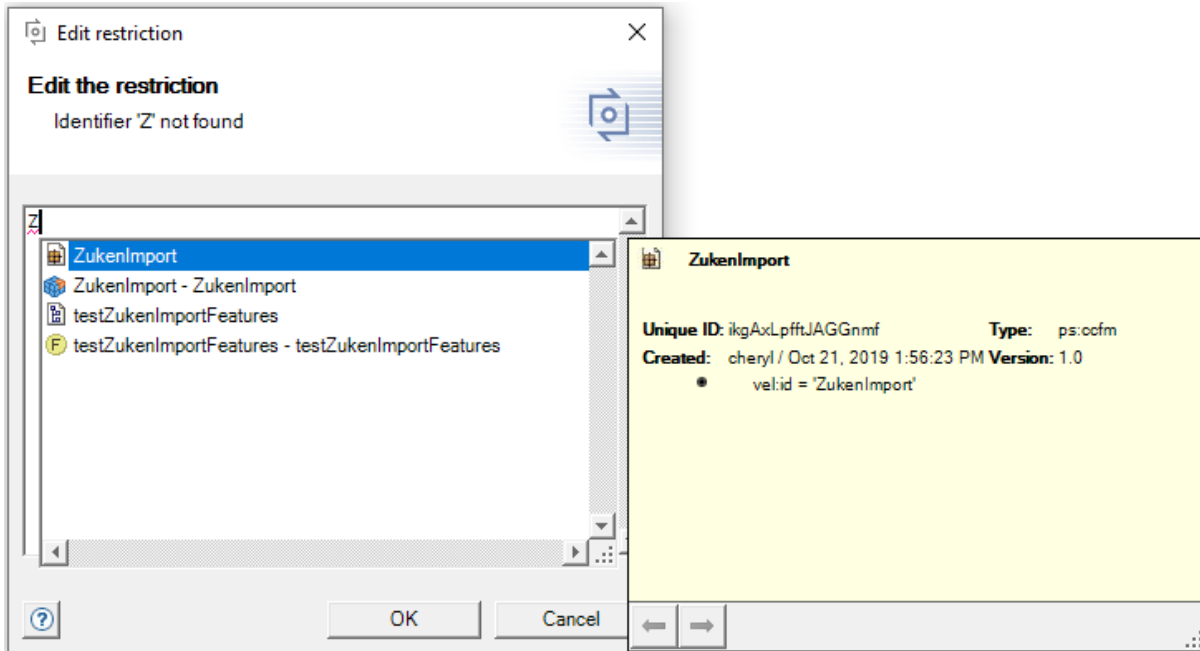
Working with the Restriction Editor

To add and maintain the variability information of an Zuken CR-8000 project more efficiently and less error-prone, the Integration provides a Restriction Editor.

You can open it by double clicking on the pvRestriction column in the Design Variation Items grid. The Restriction Editor features auto completion and syntax highlighting (see [Figure 15, "Using the Restriction Editor"](#)). You can

automatically complete words by pressing CTRL + space. If more than one word is possible, a list containing possible keywords and features is displayed. For auto completion the loaded pure::variants models are evaluated. Thus, it is recommended to load a pure::variants model before using the Restriction editor (see [the section called “Connecting with pure::variants Models”](#)).

Figure 15. Using the Restriction Editor

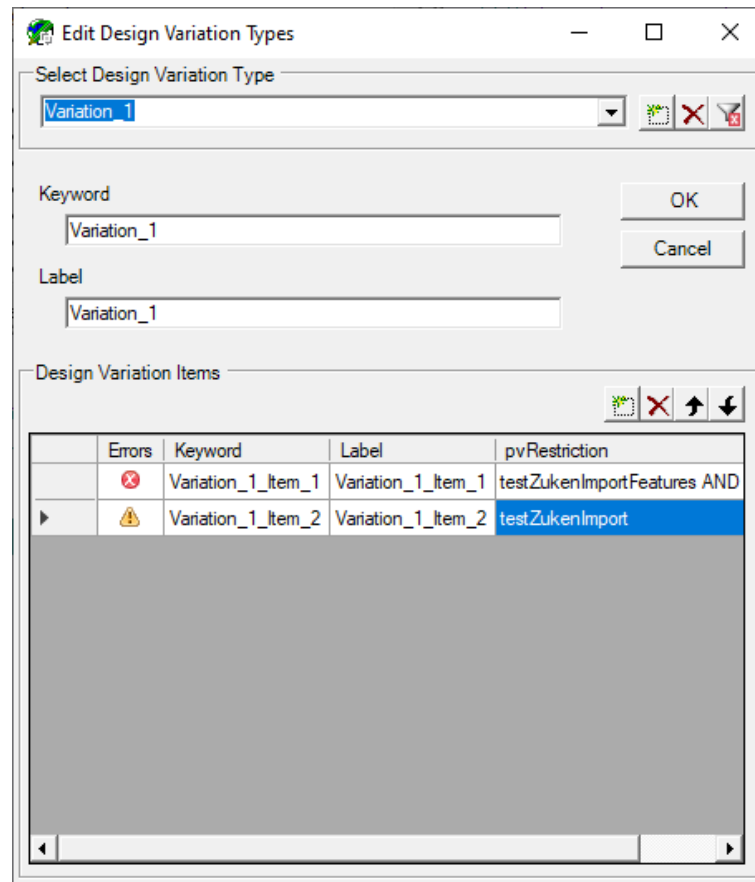


Error Check

When you are done editing the pure::variants restriction and click OK, the entered expression is checked for errors. Errors in pvSCL scripts are reported if the script's syntax is not pvSCL compliant, or if an element is unknown based on the loaded pure::variants models. Unknown elements are highlighted in red.

Finding Problems in the Variations

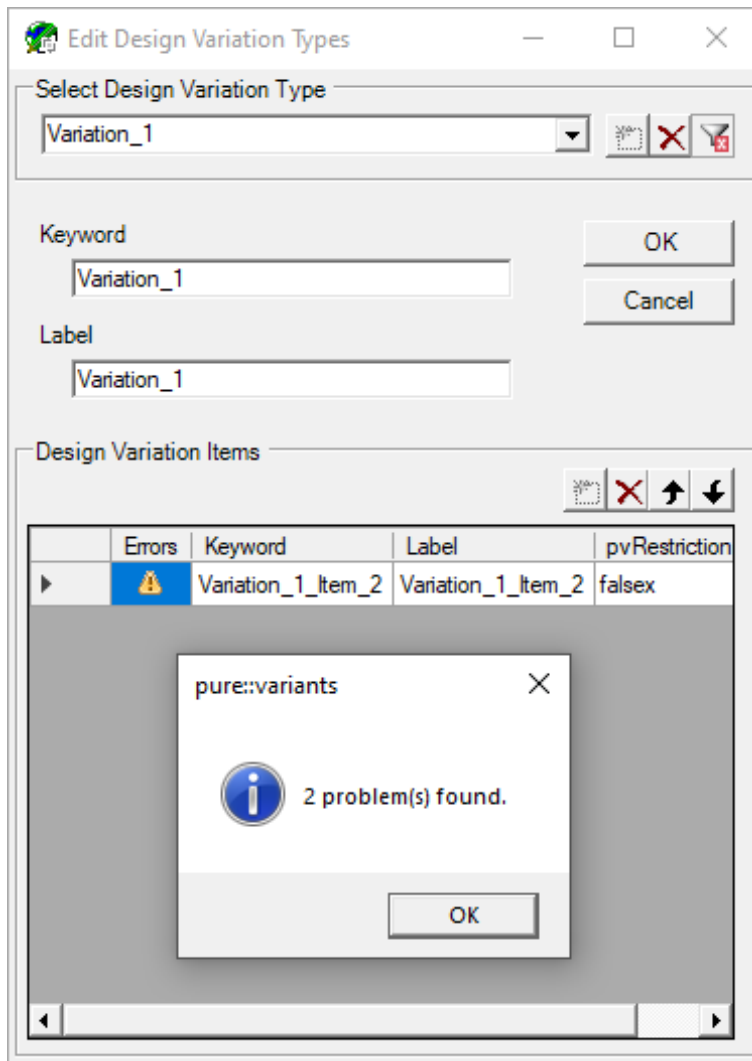
Error visualizations indicate where errors in variability information exist (errors in the pvSCL expression of pure::variants restrictions). To be able to view Problem Visualizations, the [Figure 14, “Edit Design Variation Dialog”](#) has to be loaded. The problems are shown under the "Problems" column in the grid. There are two types of problems: Elements with errors are highlighted in red, whereas elements with warnings are highlighted in yellow. [Figure 16, “Problem Visualization”](#) shows an example for the error and warning visualization. The item Variation_1_Item_1 contains the restriction "testZukenImportFeatures AND", which is a syntactic error, since the restriction is not completed. Thus, the class is highlighted in red. Apart from problems, the visualization also highlights all elements with multiple pvRestriction restrictions, since multiple pvRestrictions restrictions on one element are not allowed, and cause the transformation to fail.

Figure 16. Problem Visualization

The item Variation_1_Item_2 contains the restriction "testZukenImport", which is semantically incorrect, since the loaded feature model does not contain a feature "testZukenImport" (but it does contain a feature "testZuken-ImportFeatures"). You can easily identify this typing error through the problem visualization.

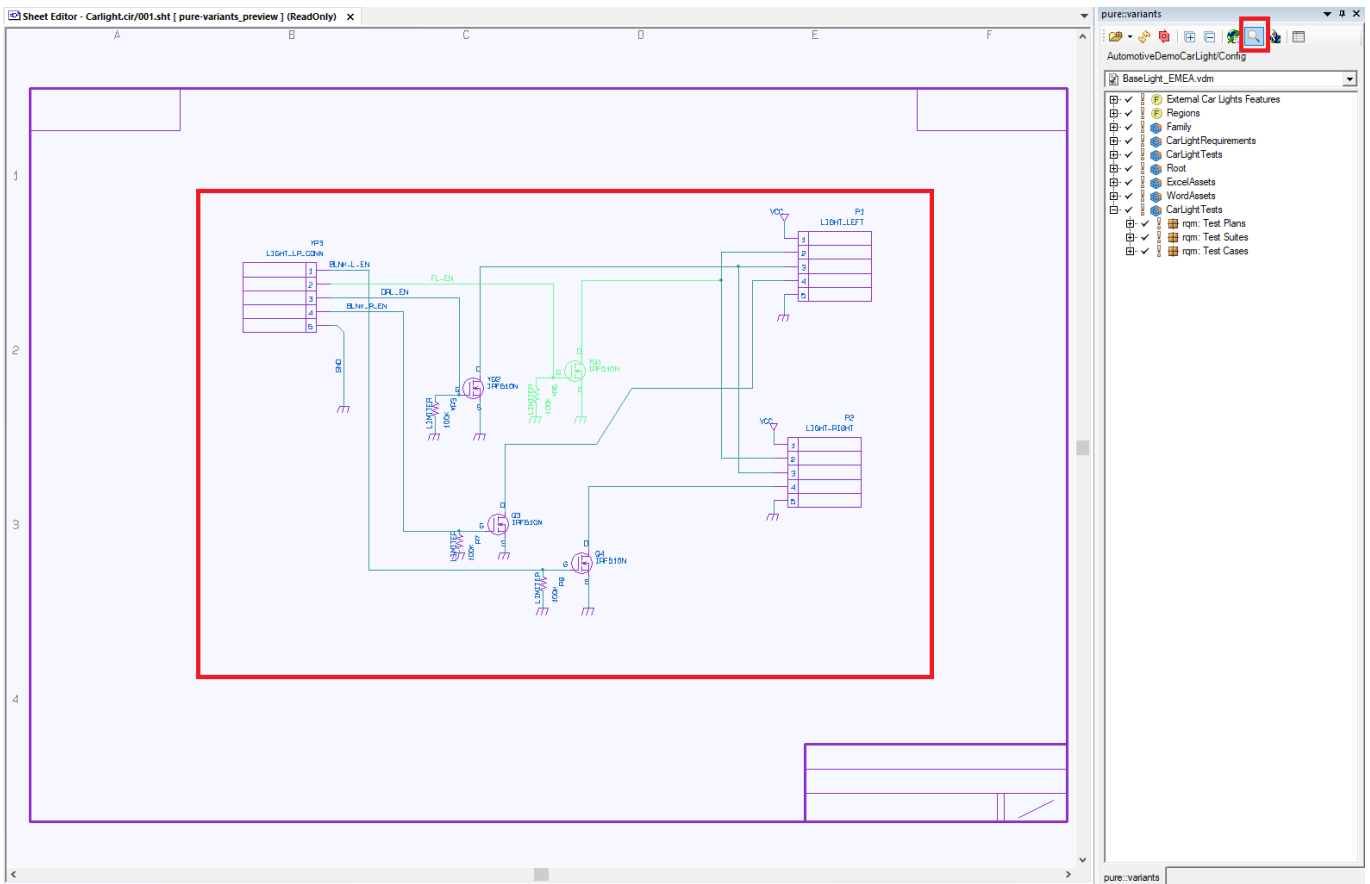
Filtering the Variations

Filtering the variations will show you only the variation types and items that have pvsc1 problems in the pvRestrictions. To filter the variations click in the [Figure 14, "Edit Design Variation Dialog"](#). Now a filtered view and a dialog with the number of problems will be visible as shown in [Figure 17, "Filtered Variations"](#)


Figure 17. Filtered Variations

Preview Destination of the Current Variant

Preview can help understand whether the added variability information produces the correct output. Through variant visualizations you can preview the results of a transformation with pure::variants. For this visualization you need to load a variant model, from which the visualizations are generated. To preview the destination click on in the Integration window. This creates a temporary destination which previews the result of the transformation. When you see the visualization, all elements that would be present in the current variant will be highlighted as shown in [Figure 18, "Preview Destination"](#). When this button is unchecked the destination is removed.

Figure 18. Preview Destination

Update Destination of the Current Variant

Updating the destination either creates a new destination or updates an existing destination. It also creates a visualization of the updated destination. To update the destination, click on  in the Integration window. This creates a permanent destination and produces the same result as the transformation.

By default, the name and label of the destination is equal to the name of the variant model. To define a special destination name and label please refer to the section [Section 2.4, “Defining New Destinations for the Zuken Design Model”](#). When you see the visualization, all elements that are present in the current variant will be highlighted. When this button is unchecked the selected destination is switched back to the default destination. The newly created destination will be still present.

Troubleshooting


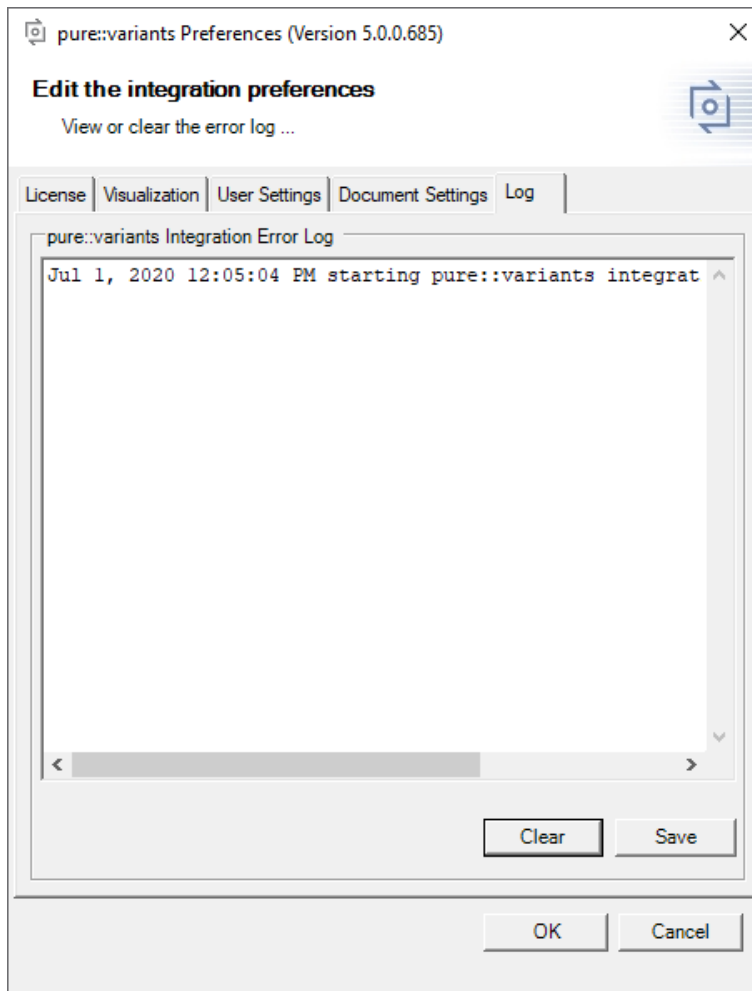
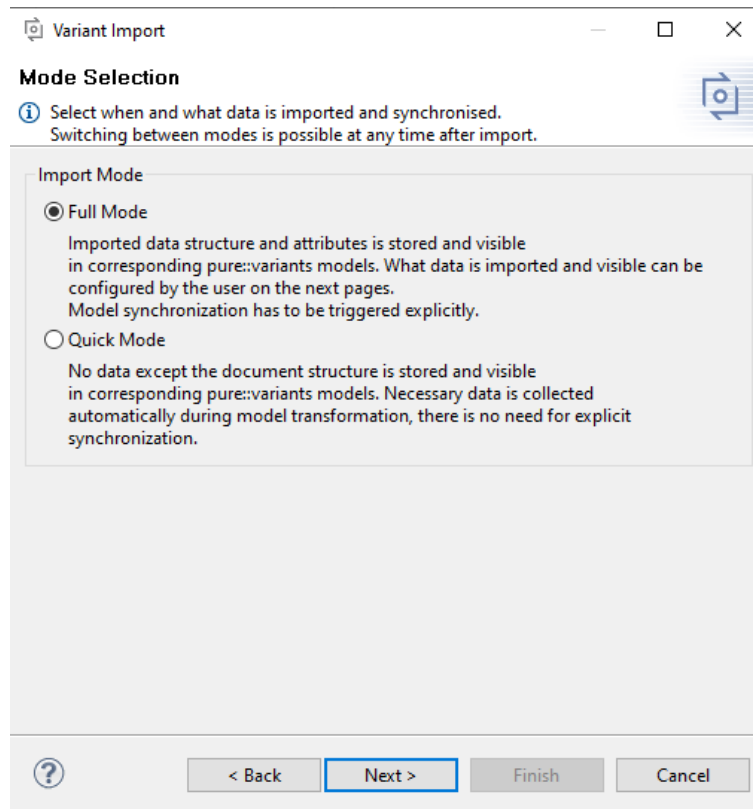
If the Integration does not behave as expected, it may be useful to check its log file. You can find it in the Integration preferences () on the **Log** tab (see [Figure 19, “Preferences Dialog Log Tab”](#)). It also enables you to save it to your disk or clear the contents of the log file.

Figure 19. Preferences Dialog Log Tab

2.3. Importing Variation Information from Zuken Files

In order to be able to control and transform variant specific Zuken design models, the SDM file from Zuken needs to be imported into pure::variants. After the import, depending on the import method, either the file itself (Quick Mode) or the variability information (Full Mode) will be represented as family models. Additionally, when using Full Mode there is also the possibility to import destinations contained in SDM files as variant models..

The import wizard for Zuken can be opened by calling the menu entry **Import...** in the **File** menu or by calling the context menu entry **Import...** in the **Variant Projects** view. In the wizard, select **Variant Models or Projects** in the **Variant Management** folder, press **Next**, select **Import Zuken Variation Resource File** and press **Next** again. On the next page ([Figure 20, “Zuken import mode alternatives”](#)), the Import Mode needs to be selected.

Figure 20. Zuken import mode alternatives

In that wizard page (see [Figure 21, “Zuken file import wizard in full mode”](#)), the target project or folder and the location of the Zuken SDM file shall be defined.

Figure 21. Zuken file import wizard in full mode

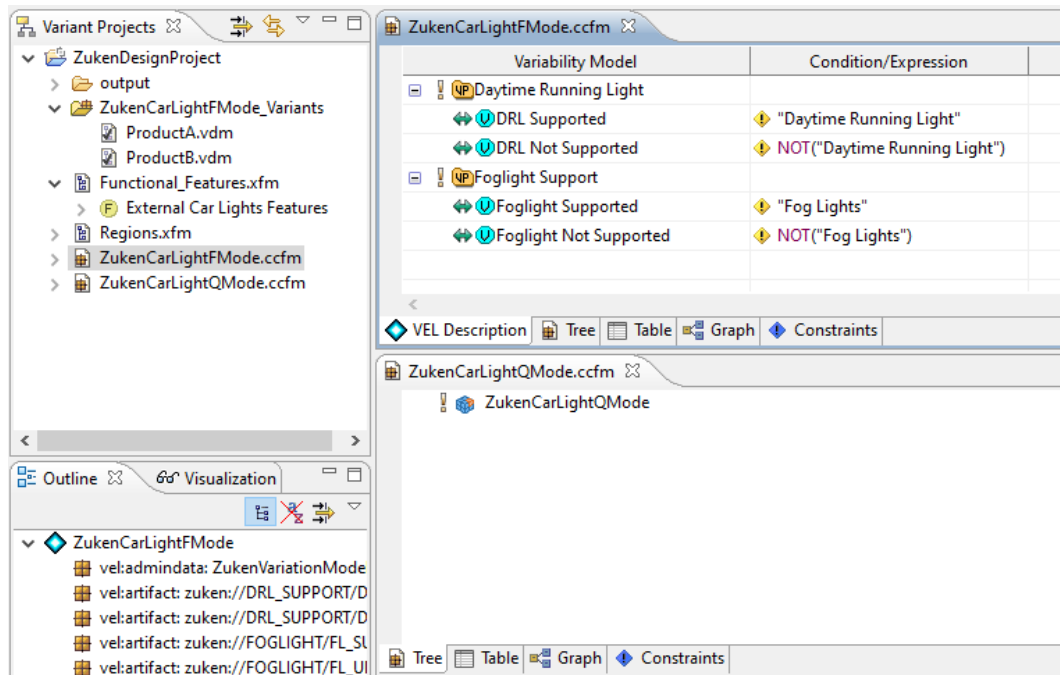
The family model stores the reference to the SDM file in different ways, that needs to be decided during the first import by the user. The user can select between absolute or relative path handling options using the listed alternatives under **Import path handling** section.

In case of Full Mode, if the option **Import contained Zuken destinations as variant models** is activated, during the import a pure::variants configuration space with the name <family model name>_Variants is created, if it does not exist. In that configuration space, for each Zuken destination, a variant model is created, containing user selections based on their selected variation items. The name of the created variant model is equal to the name of the destination. If a variant model with that name already exists, it will be overwritten, but additional variant models added by the user in that configuration space before remain intact.

During Full Mode import, a VEL family model with the given name is created describing

- each Zuken variation type as a VEL variation point
- each Zuken variation item as a VEL variation.

The imported family model does not contain any mapping to features. This has to be done manually.

Figure 22. Zuken Family Models as results of full and quick modes

In case of Quick Mode, a simple family model is created containing only a simple element as a representation for the SDM file (Figure 22, “Zuken Family Models as results of full and quick modes”).

Both models can be re-synchronized with the SDM source using the option **Synchronize the current editor model** from the toolbar.

2.4. Defining New Destinations for the Zuken Design Model

New destinations for the Zuken design model can be defined by creating new variant models in the configuration space containing the imported family model.

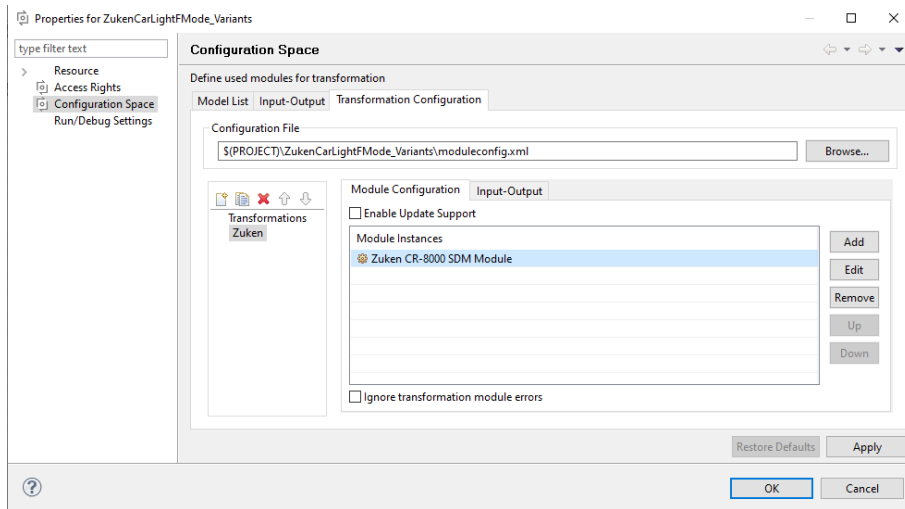
By default, the name and label of the destination is equal to the name of the variant model. To define a special destination name or label differing from the name of the variant model, following model attributes can be added and set in the variant model:

zuken:destinationname The name of the Zuken destination

zuken:destinationlabel The name of the Zuken destination

2.5. Creating Zuken Destinations by Transformation

The creation of new or updated Zuken destinations based on a set of variant models is done by a special Zuken transformation. This can be configured by adding the transformation module **Zuken CR-8000 SDM Module** to a transformation configuration of the corresponding configuration space (Figure 23, “Set transformation module”).

Figure 23. Set transformation module

As a result, an updated SDM file is created directly in the output directory defined by the transformation configuration. For each Zuken family model the referenced SDM file is updated. In case multiple variants are transformed at the same time targeting the same SDM file, the same file is updated multiple times. That is, if the Zuken SDM file should contain more than one destination, the corresponding transformation configuration have to be launched for more than one variant model at one time.

3. Known Restrictions

None.