

Using the Code Assistant

This chapter covers the following topics:

- About the Code Assistant
 - Code Template
 - Opening the Code Assistant
 - Code Introspection
 - Generating the Code
 - Editing the Adapter Code
 - Saving Changes
 - Changes from the Outside
-

About the Code Assistant

The Code Assistant only applies to PAGE layouts.

When developing page layouts in Java, automatic code modifications do not occur. For example, when you set the `valueprop` property in a FIELD control, the `set ()` and `get ()` methods are not automatically created in the underlying Java adapter class. Using the Code Assistant, however, you can generate code for all properties and methods which are explicitly referenced in the page layout. The compilation of the modified adapter class itself is done in your Java development environment (for example, in Eclipse).

When developing page layouts in Natural, the corresponding Natural code for the properties and methods is automatically generated when you build a user interface component, when you build your project in Eclipse, or when you simply save a modified layout in Eclipse and the **Build Automatically** command in the **Project** menu is activated. Therefore, the Code Assistant is not required for NATPAGE layouts and the corresponding menu commands are disabled.

Code Template

Before you open the Code Assistant, make sure that the `model` property of the page contains the correct adapter class. The default entry for this property is "DummyAdapter".

The Code Assistant looks in the source directory for the adapter class you have set with the `model` property. If the class cannot be found, the Code Assistant loads a code template.

The code template is stored in the directory `<your-webapplication>/cis/config` . For NaturalONE, this is usually `<webcontainerinstalldir>/webapps/cisnatural/cis/config`. Otherwise, it is usually `<webcontainerinstalldir>/webapps/cis/cis/config`.

The name of the code template is *adapterTemplate.java*. You can change this template according to your needs.

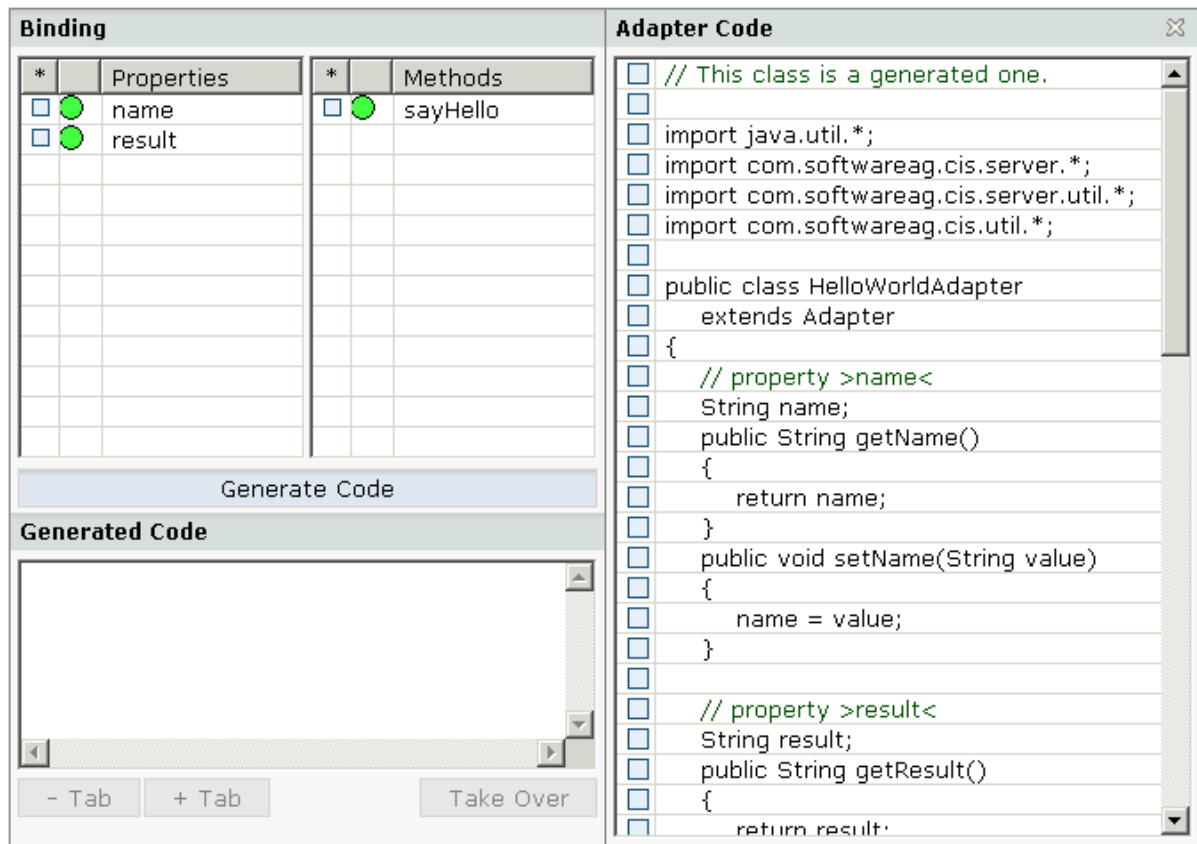
Opening the Code Assistant

In order to read and write source code, the Code Assistant needs to know where the Java sources are kept. This information is stored in the file *ciseditorconfig.xml* of each user interface component. When working with Eclipse, a *src* subdirectory is automatically created in each user interface component and applied as a Java source directory to your Eclipse project. The corresponding settings are automatically saved to the file *ciseditorconfig.xml*.

▶ To open the Code Assistant

- From the **Ajax Developer** menu, choose **Open Code Assistant**.

When the Java source directory can be found, the Code Assistant is shown in the preview area. For example:



Code Introspection

The binding area at the top left of the Code Assistant lists all properties and methods that are used in the XML layout. Colored dots are shown in front of each property and method. The color of the dot indicates whether the Code Assistant could find the corresponding coding within your source code.

Introspection is done by reading and interpreting the code. Comments that highlight properties or methods are not necessary.

Possible dot colors for properties:

Green	The property's <code>get</code> and <code>set</code> methods are available.
Yellow	The <code>set</code> method is required but could not be located. The <code>get</code> method is fine.
Red	The <code>get</code> method could not be found.

Possible dot colors for methods:

Green	The method is available.
Red	The method could not be found.

When you select a property or method in the binding area, the properties are shown in the **Properties** view.

In addition, when code has already been generated for the selected property or method (see below), the corresponding line is automatically selected in the adapter code which is shown at the right of the Code Assistant.

Generating the Code

The properties and methods which cannot be found in the adapter code are indicated by red dots.

When you generate code, a framework for the property or method is generated in the adapter code.

To generate code

1. Select the property or method for which you want to generate code.

You can also use the **SHIFT** or **CTRL** key to select multiple entries.

2. Choose the **Generate Code** button.

The generated code appears in the area below this button.

Check whether you really want to insert these statements into your source code. You can modify or delete the code. Any code that you manually enter here will be taken over.

3. Optional. Choose the **+Tab** or **-Tab** button to add or remove a tab at the beginning of the code. This applies to all code which is currently shown in the **Generated Code** area.
4. In the **Adapter Code** area on the right, select the line below which you want to insert the code.
5. Choose the **Take Over** button to insert the code below the selected line.

The color of the dot changes from red to green. This indicates that the corresponding methods are now available in the adapter code.

6. Save your changes.

Note:

It is recommended that you save your changes each time you switch from the Code Assistant to another tool which may be shown in the preview area, and vice versa.

Editing the Adapter Code

You can edit the adapter code in different ways:


- Editing the Adapter Code in the Java Editor
- Editing the Adapter Code Directly in the Code Assistant

Editing the Adapter Code in the Java Editor

If you want to apply complex changes to the adapter code, it is recommended that you do this in the Java editor.

Important:

Be sure to close the Code Assistant before you open the adapter code in the Java editor.

 **To invoke the Java editor**

1. In the Navigator view, select the page layout (that is, the XML file) for which you want to change the adapter code.
2. Invoke the context menu and choose **Ajax Developer > Open Java Adapter**.

Or:

When the Layout Painter is still active, open the **Ajax Developer** menu and choose **Open Java Adapter**.

Editing the Adapter Code Directly in the Code Assistant

Instead of invoking the Java editor, you can also apply smaller changes to the adapter code directly in the Code Assistant, while generating the code for the properties and methods.

 **To edit the adapter code in the Code Assistant**

1. Select one or more lines in the **Adapter Code** area.
To select more than one line, use CTRL or SHIFT.
2. Invoke the context menu and choose one of the following commands:

Command	Description
-Tab	Removes a tab from the beginning of the selected line(s).
+Tab	Adds a tab to the beginning of the selected line(s).
Cut	Cuts the selected line(s).
Copy	Copies the selected line(s).
Paste	Only available after a cut or copy operation. Pastes the cut or copied line(s) below the currently selected line.
Remove	Removes the selected line(s).

Saving Changes

You save the adapter code using the standard Eclipse functionality for saving.

Notes:

1. When editing the adapter code using the Code Assistant, it is recommended that you first close the Code Assistant before you close the Layout Painter.
2. After saving your changes to the adapter code, choose the **Refresh Preview** command so that you can see the changes in the preview area.

Changes from the Outside

Each time you preview the layout, the Code Assistant checks whether the file has changed on the file system. Therefore, the file attribute "last modified" is examined (not the file content). If the Code Assistant notices a file change, it checks whether it is holding own unsaved changes. If not, the class code is reloaded from the file system. If you have done changes in parallel (within your Java development environment and the Code Assistant) a dialog appears. Choose either to reload the file from disk (and lose the changes you have done within the Code Assistant) or to keep the Code Assistant changes (and lose the outside changes on the next save).

Note:

You have to take care that the following case never occurs: the file has changed on file system and the Code Assistant has its own unsaved changes. Of course, it is reasonable to open the class both in your Java development environment and in the Code Assistant. It is recommended that you save or discard changes before you switch from the Code Assistant to your Java development environment (or vice versa). Otherwise, you will lose one of your changes.