

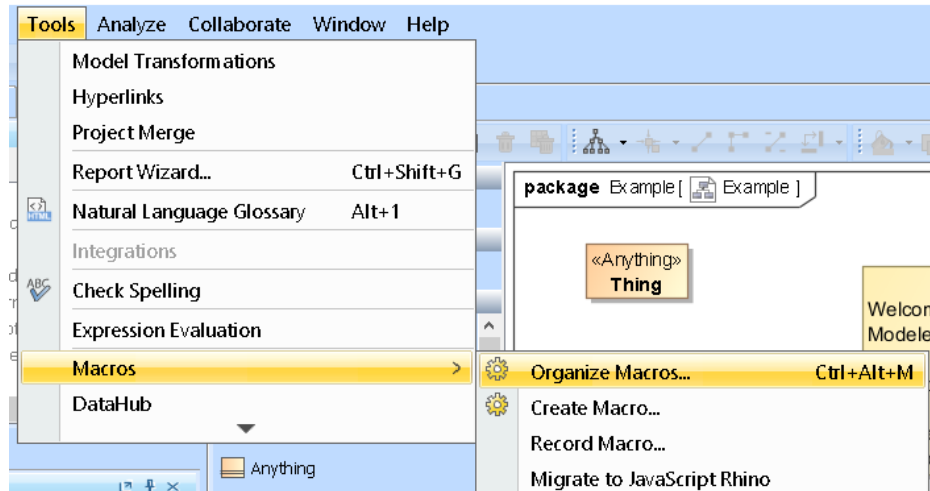
Migrate Older Models to Use Relative IRIs

AVAILABLE FROM 19.0

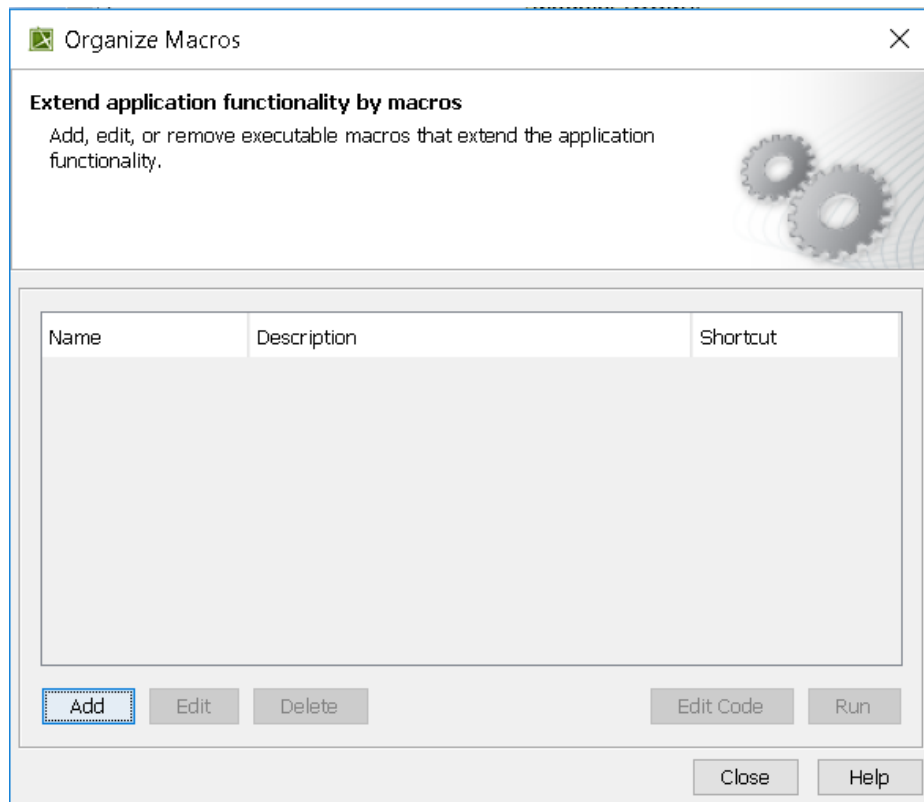
Migrating older models to use relative IRIs improves model refactoring. This can be done using the [Macro Engine](#). The modeling tool comes with the macro file installed in the tool's directory. Instructions on how to find the file are described in **step #4** of how to add the macro.

To add the macro

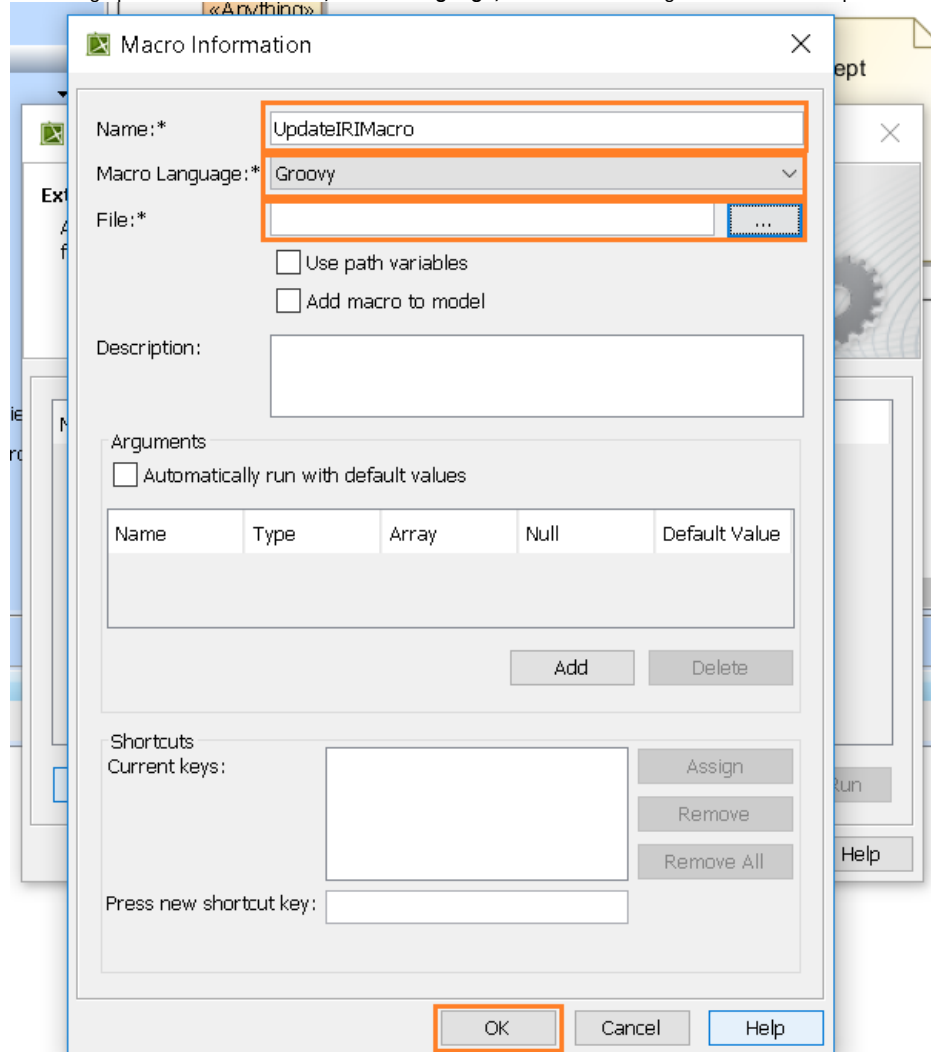
1. In your Concept Modeling Project, click **Tools > Macros > Organize Macros...**, or use the shortcut key **Ctrl+Alt+M**.



2. The **Organize Macros** dialog opens. Click **Add**.



3. The **Macro Information** dialog opens. Fill out the **Name**, **Macro Language**, and **File** containing the macro's description.



4. To find the appropriate file, click on the three dots at the end of **File** field. Your local directory opens. Find the application folder, and select **data > Concept Modeler > scripts > UpdateIRI.groovy**.

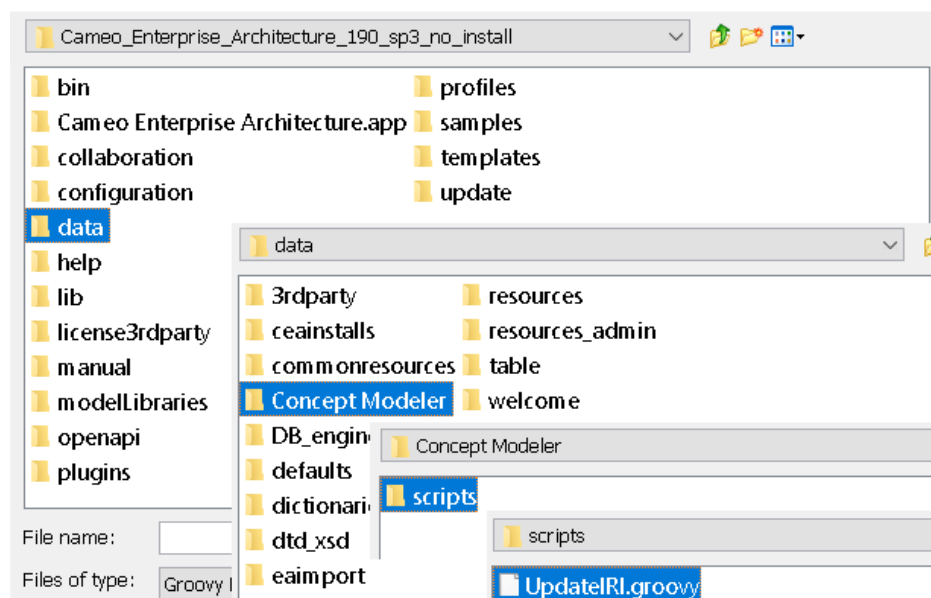


Figure 4: Showing the steps to find the .groovy file

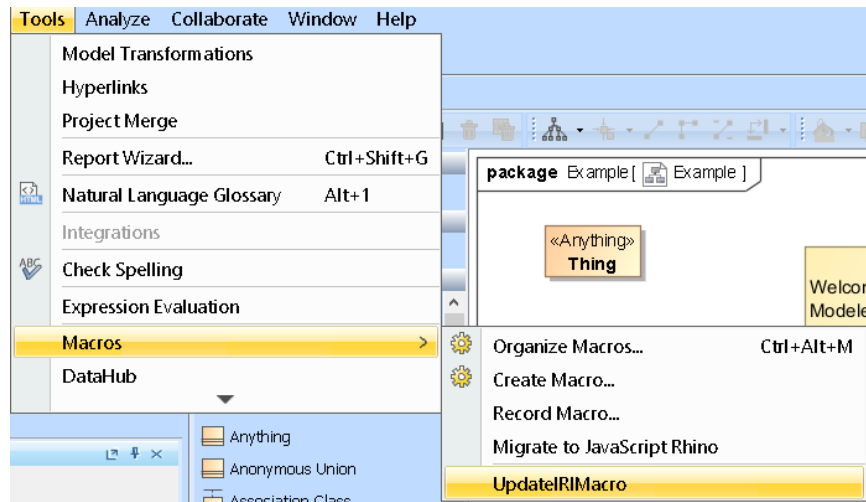
5. Now that you have selected the file, click **OK**. You should now see the macro added in the **Organize Macros** dialog.
6. Click Close.

Macro Language

The **Macro Language** is the macro's file extension. The modeling tool supports **BeanShell**, **Groovy**, **JRuby**, **JavaScript**, **JavaScript Rhino**, and **Jython**.

To execute the macro

1. Click on **Tools > Macros > Macro Name** e.g. *UpdateIRIMacro*.

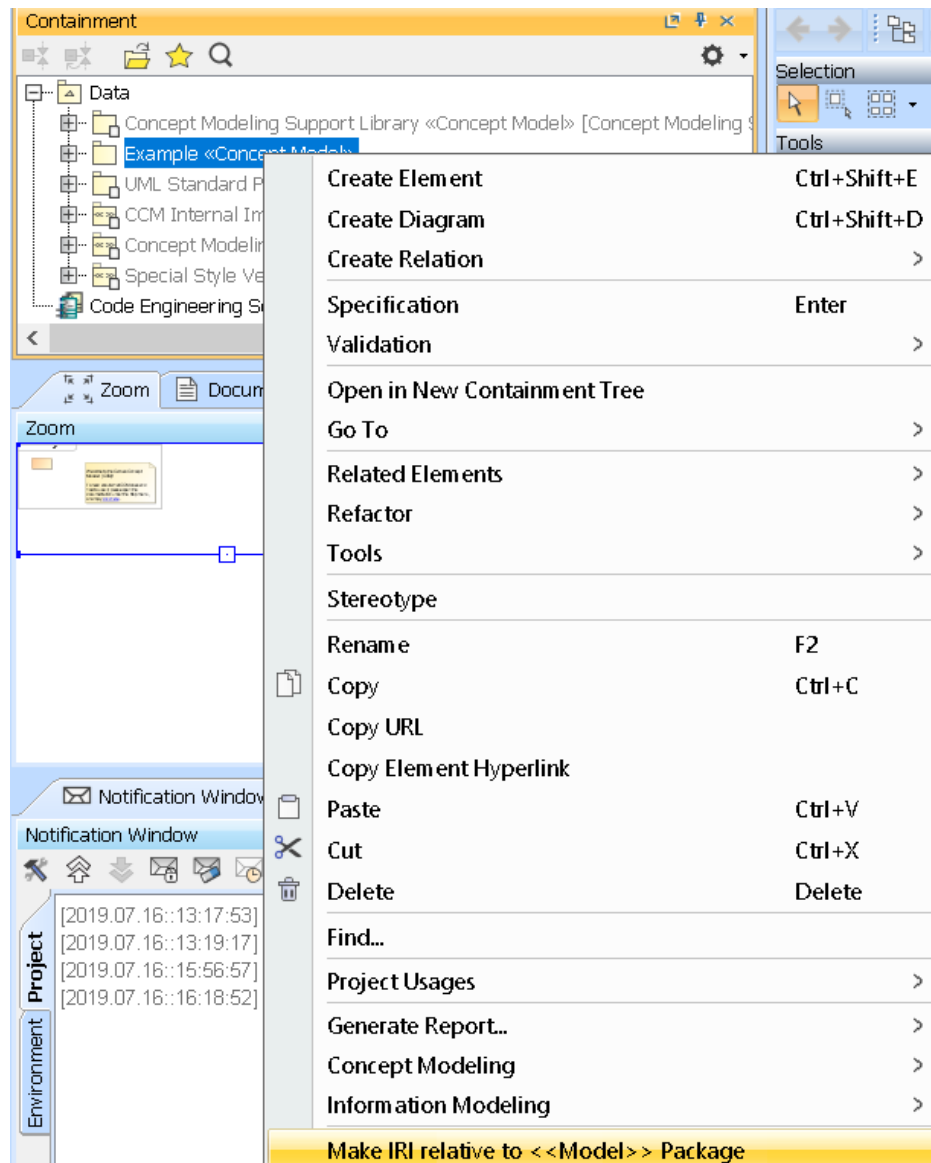


2. The **Notification Window** will notify you when the macro was executed.

The macro *UpdateIRIMacro* has been executed.

To use the macro for your existing project

1. Right-click on the model on which you would like to update IRIs.



2. A dialog will open asking you where to save the results in .csv format.
3. After selecting the file, the progress bar shows the progress of the migration.
4. The notification window notifies you when the migration is done, and where the results are stored.

Update IRI results saved to: 'C:\Users\ \Desktop\iridemo1.csv'

5. Find the Update IRI results in your designated location, and view the CSV file showing which elements were looked at and updated. The CSV file describes the "Element ID", "Old IRI", "New IRI", and "Updated".



If you're working with a TWC project...

- For Teamwork Cloud (TWC) projects, the script will try and lock an element before updating its IRI value. For any element that should be updated and cannot be locked, a message will be shown in the **Notification Window**: "Unable to lock and update IRI for '%s' due to an element being locked by another user".
- For TWC projects, Property typed elements cannot be locked, so their owner will be locked instead.



If your element has no IRI...

- If current element has no IRI value, then no IRI value will be created for it.