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.

	Тоо	<mark>Is</mark> Analyze Collaborate Window Help			
		Model Transform ations			
		Hyperlinks			
		Project Merge	Ô		F
		Report Wizard Ctrl+Shift+G		package Example [📑 Example]	
d	្ឋា	Natural Language Glossary Alt+1			
		Integrations		«Anything»	
d T	ABS	Check Spelling		Thing	
- 31		Expression Evaluation	^	Welco	
E		Macros >	ŝ	Organize Macros Ctrl+Alt+M	
		DataHub	÷	Create Macro	u
٦		• 		Record Macro	e
		Anything		Migrate to JavaScript Rhino	1

2. The Organize Macros dialog opens. Click Add.

Image: Image					
	xtend application functionality by macros Add, edit, or remove executable macros that extend the application functionality.				
Name	Description	Shortcut			
Add Edit	Delete	Edit Code	Run		
		Close	Help		

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

-	🔯 Macro Inform				×	ept	
Ŕ	Name:*	UpdateI	RIMacro			×	
Ext	Macro Language:	* Groovy			~		
4	File:*						
f		Use	path variables				
		Add	Add macro to model				
	Description:						
Ν	Arguments						
	Name	Туре	Array	Null	Default Value		
				Add	Delete		
	Shortcuts				Assign		
	Shortcuts Current keys:				Assign	tun	
					Remove		
	Current keys:					Run Help	
		cut key:			Remove		
	Current keys:	cut key:			Remove		

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 > UpdatelRI.groovy.

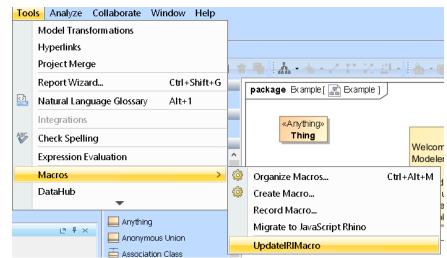
Cameo_Enterprise_4	Architecture_190_sp3_no_install 🛛 🗸 🤌 😥 🖽 🗸
📜 bin	profiles
📒 Cameo Enterprise	Architecture.app 📒 samples
📒 collaboration	Etemplates
Configuration	📒 update
L data	📜 data 🗸 🗸
Lib	3rdparty resources
license3rdparty	ceainstalls resources_admin
📜 m anual	🦲 commonresources 📒 table
📜 m odelLibraries	📙 Concept Modeler 📒 welcom e
📒 openapi	DB_engin
📒 plugins	defaults
	📕 dictionari, 📕 scripts
File name:	dtd_xsd
Files of type: Groovy I	eaimport UpdatelRl.groovy
5. Now that you have selected the file, click Ok. Y6. Click Close.	Figure 4: Showing the steps to find the <i>.groovy</i> file You should now see the macro added in the Organize Macros dialog.

Macro Language

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

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 UpdatelRIMacro 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.

Containment		ē 平 ×	🔺 🔺 : 🖓 🛛 :		
🛋 🛤 🖆 🚖 🔍	Selection				
🖵 🗖 Data					
		oport Library «Concept Model» [Concept Modeling \$	Tools		
Example «Conce	unt hide	Create Element	Ctrl+Shift+E		
🖻 🔂 UML Standard P		Create Diagram	Ctrl+Shift+D		
🗊 - 🔄 Concept Modelir		Create Diagram			
🗄 - 🔄 Special Style Ve			>		
Code Engineering S		Specification	Enter		
<		Validation	>		
📑 🕺 Zoom 📄 Docum		Open in New Containment Tree			
Zoom		Go To	>		
mumory linealing		Related Elements	>		
		Refactor	>		
		Tools	>		
		Stereotype			
		Rename	F2		
	Ď	Сору	Ctrl+C		
		Copy URL			
		Copy Element Hyperlink			
🖂 Notification Windov	P	Paste	Ctrl+∀		
Notification Window	×	Cut	Ctrl+X		
🛠 🗞 😓 🖂 🅱 😿	ۍ ≣				
[2019.07.16::13:17:53]		Delete	Delete		
		Find			
[2019.07.16::13:19:17] [2019.07.16::15:56:57] [2019.07.16::16:18:52]		Project Usages	>		
		Generate Report	>		
u de la companya de l		Concept Modeling	>		
Environment		Information Modeling	>		
		Make IRI relative to < <model>> Package</model>			

A dialog will open asking you where to save the results in .csv format.
 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.