CrySL Visual Order Editor – User Manual (Sirius/Xtext Version)

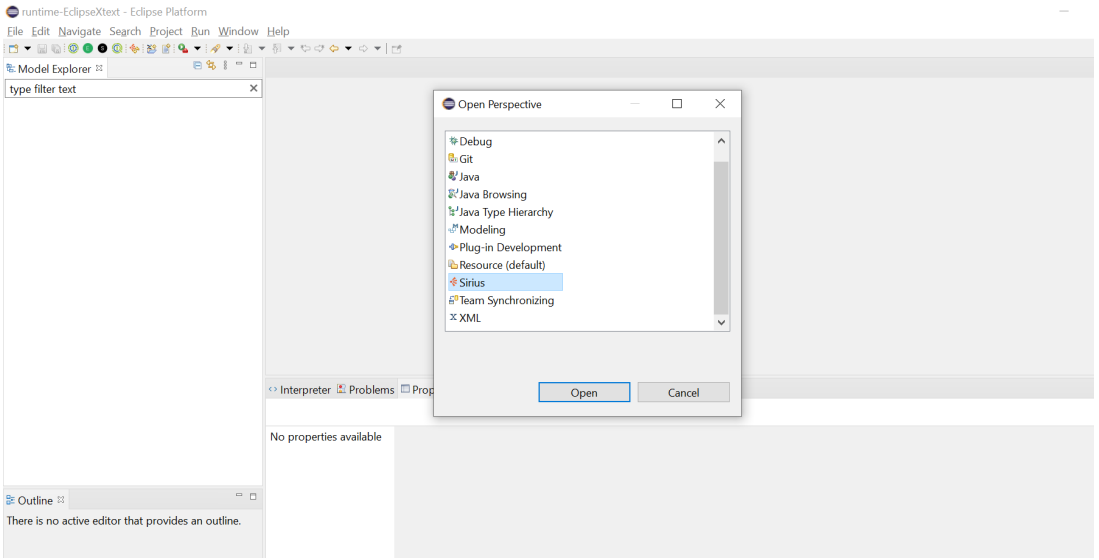
The Visual Order Editor displays the usage pattern of the class as defined in the Order expression of its CrySL rule as a state machine.

Set up

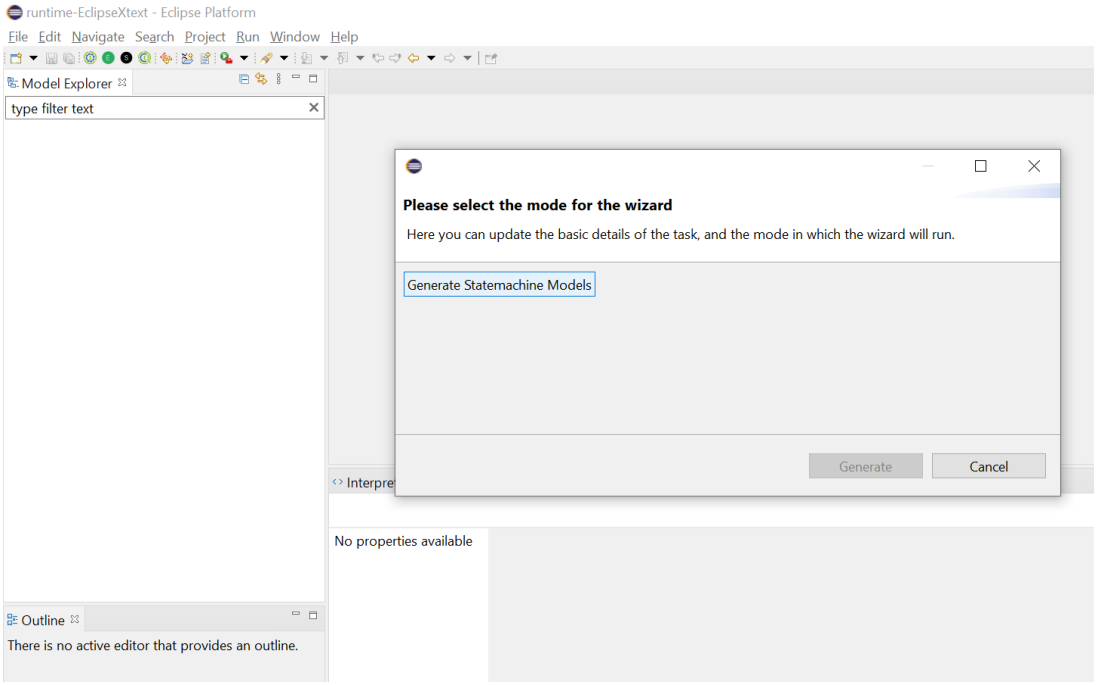
- Clone the CogniCrypt repository.
- Clone the Crypto-API-Rules repository (https://github.com/CROSSINGTUD/Crypto-API-Rules) to a folder named "git" in your home directory. This is required since the class StaxWriter which writes the configuration file needs to access this different repository and therefore accesses the paths relative to home directory by e.g. "<home-dir>\\git\\Crypto-API-Rules\\JavaCryptographicArchitecture\\src".
- Install Sirius (http://www.eclipse.org/sirius) from the Eclipse Marketplace.

Create a diagram

- Launch a new runtime from your Eclipse. Within the runtime environment, select the Sirius perspective. This opens a model explorer in the left corner.

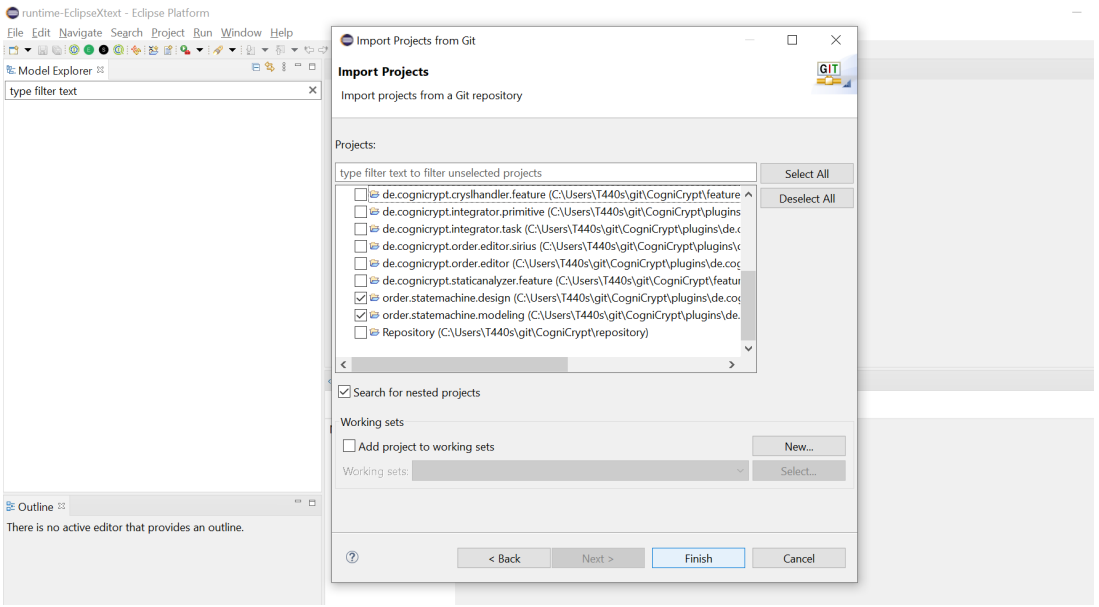


- Click on the black plugin icon "S" in the upper left corner, next to the other buttons for TaskIntegrator and CogniCrypt, then click on the button "Generate Statemachine Models", which is currently a simple button but should later be replaced by a single CrySL rule selection.

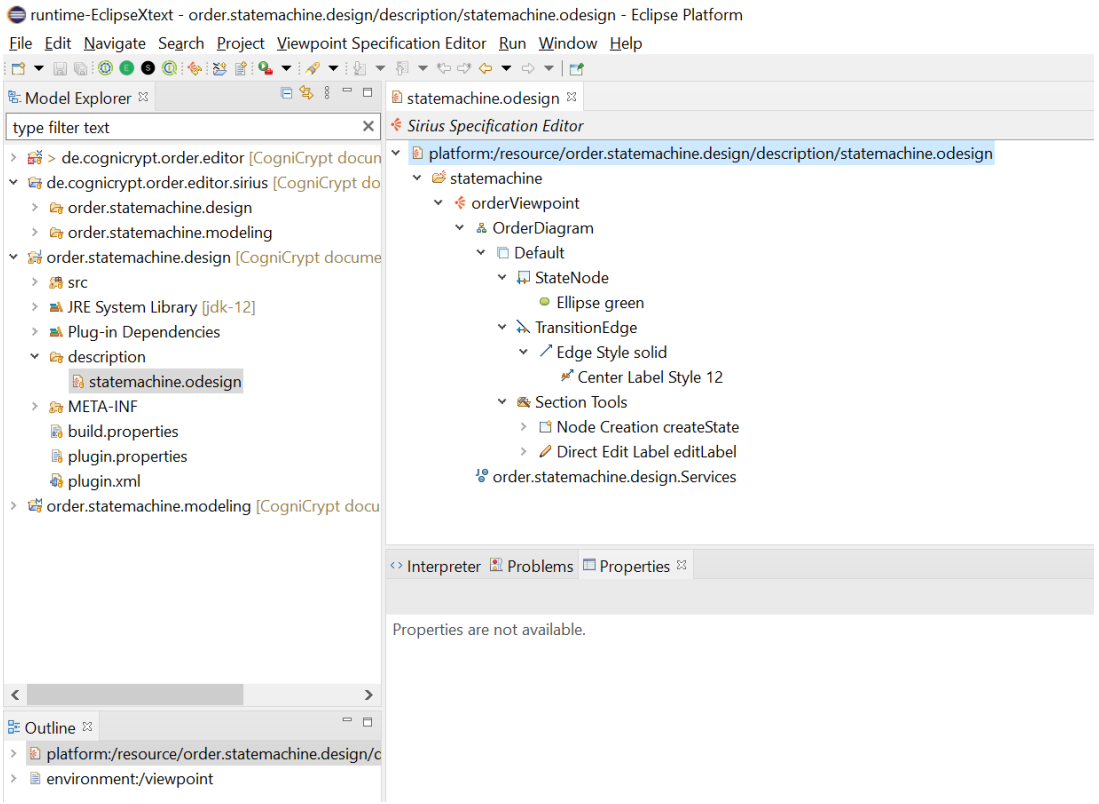


This button triggers the generation of the statemachine model resources into a separate folder, i.e., "de.cognicrypt.order.editor\output".

- Import the project "CogniCrypt\plugins\de.cognicrypt.order.editor\sirius" into the runtime environment. This folder contains the Sirius specific project files. Make sure to only add the inner projects, otherwise the projects might have referencing issues.

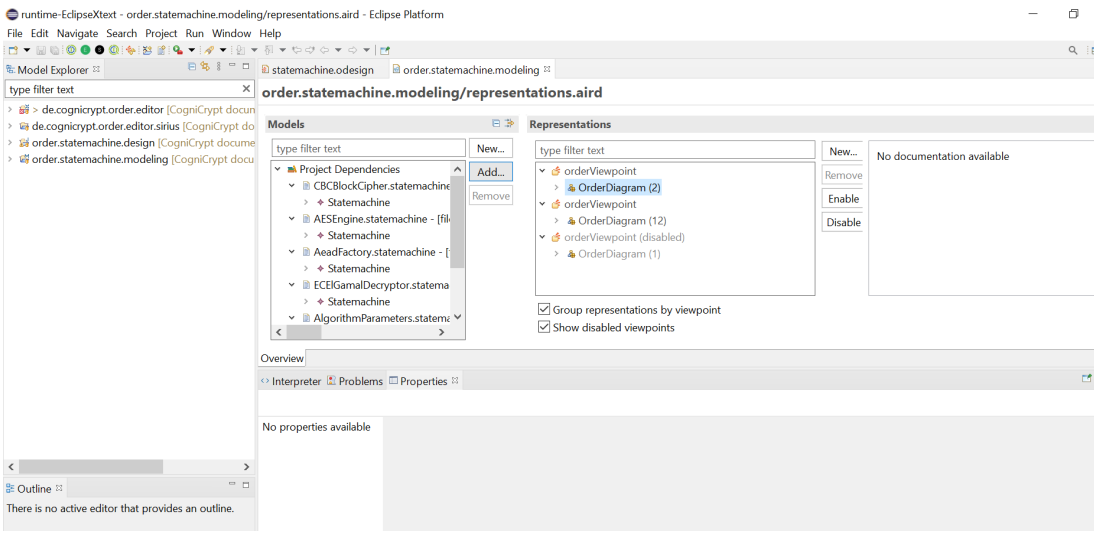


The project folder "~\CogniCrypt\plugins\de.cognicrypt.order.editor\sirius\my.project.design" is the Viewpoint Specification Project containing the .odesign file (definition of the modeling workbench).



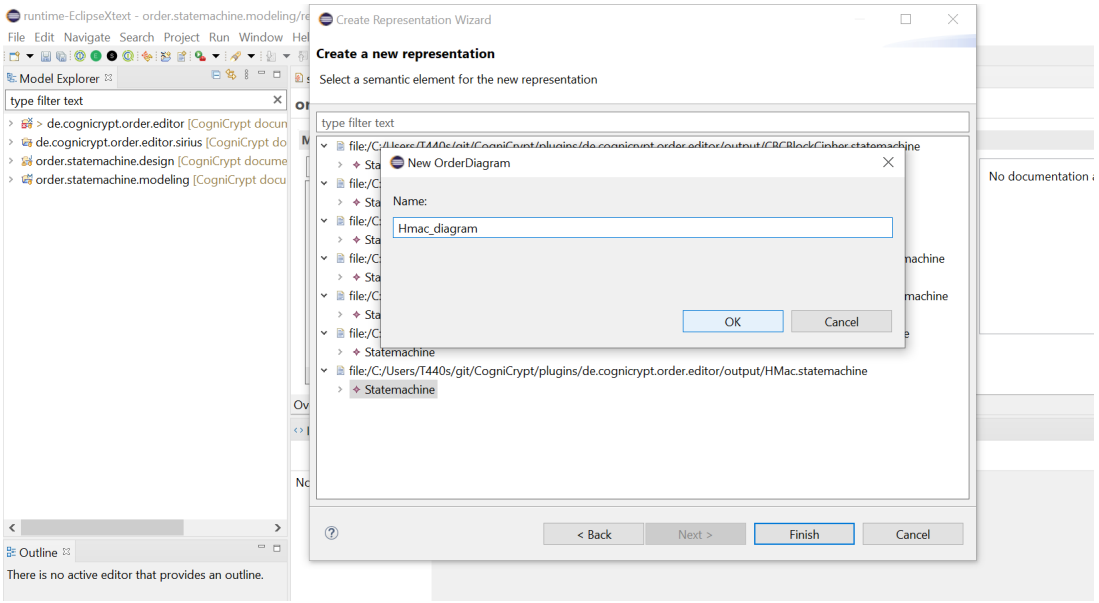
The other project folder "~\CogniCrypt\plugins\de.cognicrypt.order.editor\sirius\my.project.order.diagram.modeling" is the Modeling project containing the graphical representations created with Sirius, saved in a file representations.aird.

- To create a diagram, open the file representations.aird from the modeling project. (If it is empty, add the Statemachine.ecore model from ~\CogniCrypt\plugins\de.cognicrypt.order.editor\model\generated”). In the left corner of the new window, named "Models", click on Add > Browse File System to select a statemachine model from the plugin-relative output folder. The model will appear in the Models window. Now double click “orderDiagram” below "orderViewpoint" in the Representations window on the right.

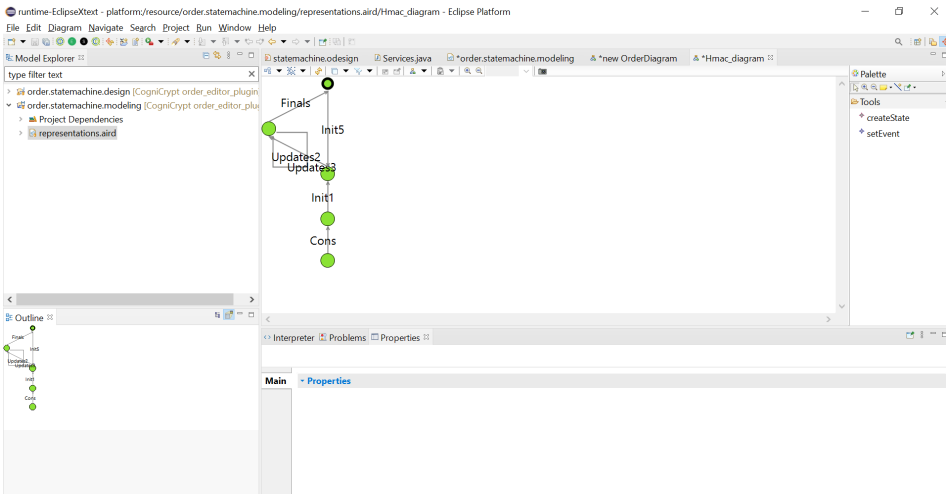


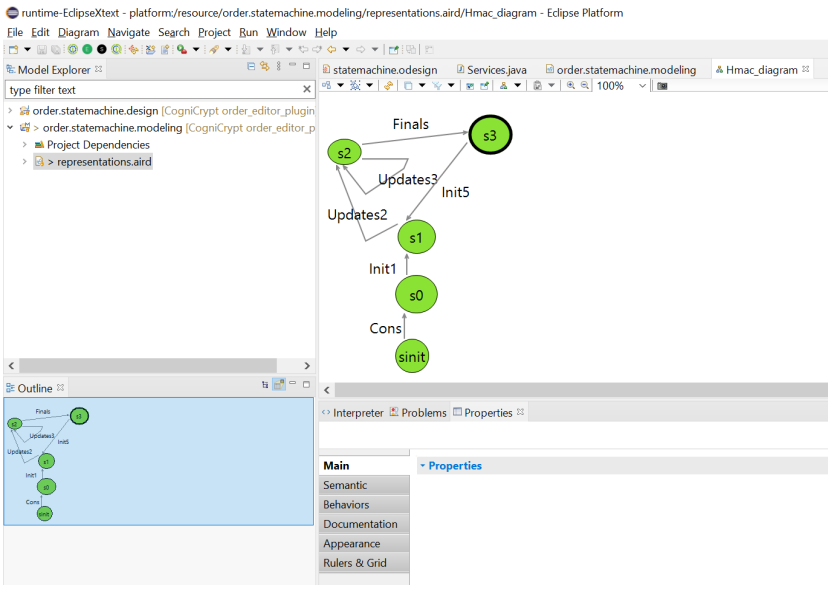
This opens a new window "Create a new representation" which allows to select a

semantic element for a new representation. Here, you can select the model you just added, click on its "Statemachine" model identifier and click on Finish. You can optionally type a custom name for the diagram.



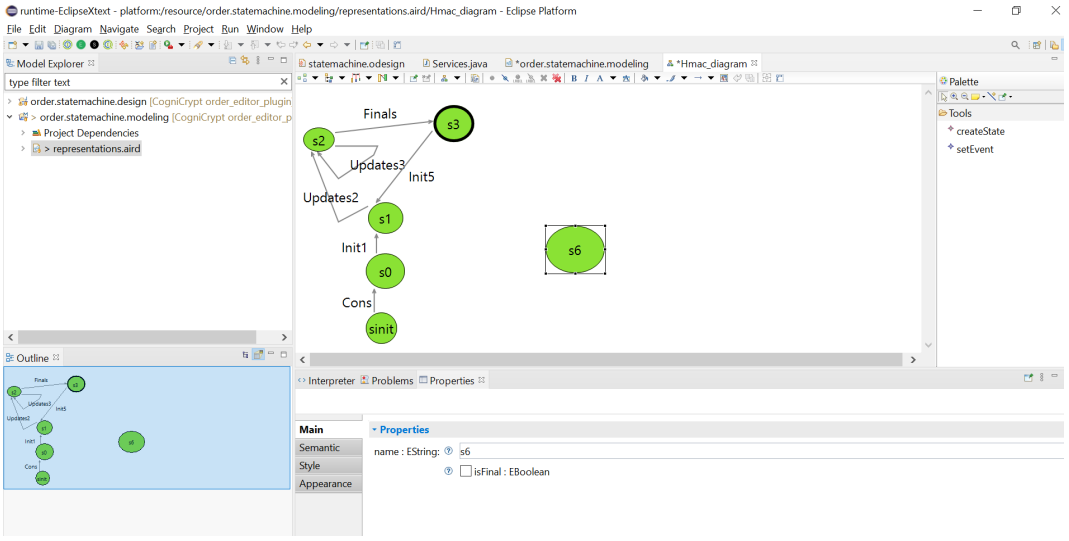
The representation is opened now and you can play around with the Sirius model editor features to enhance the representation.



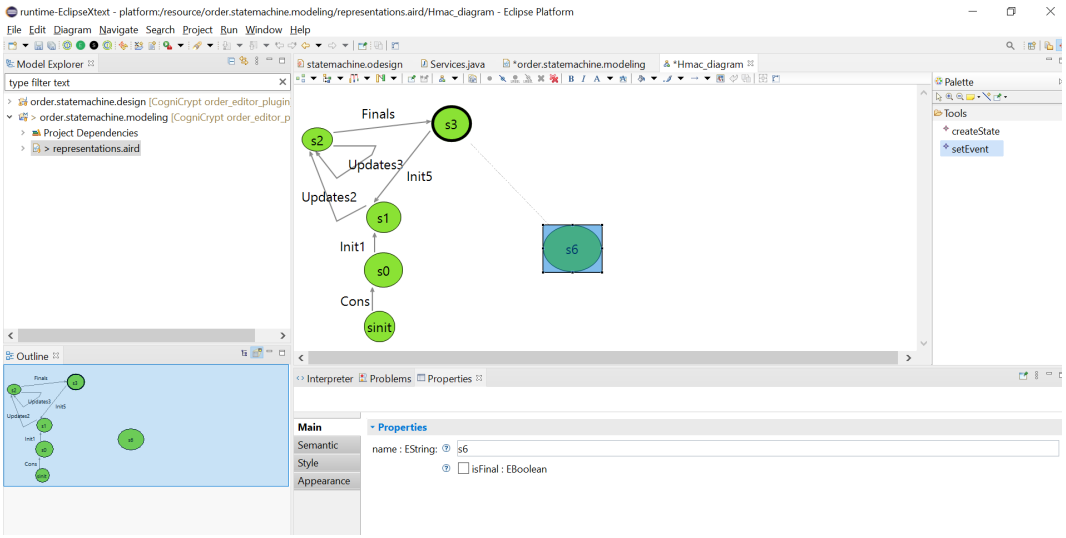


Current Features

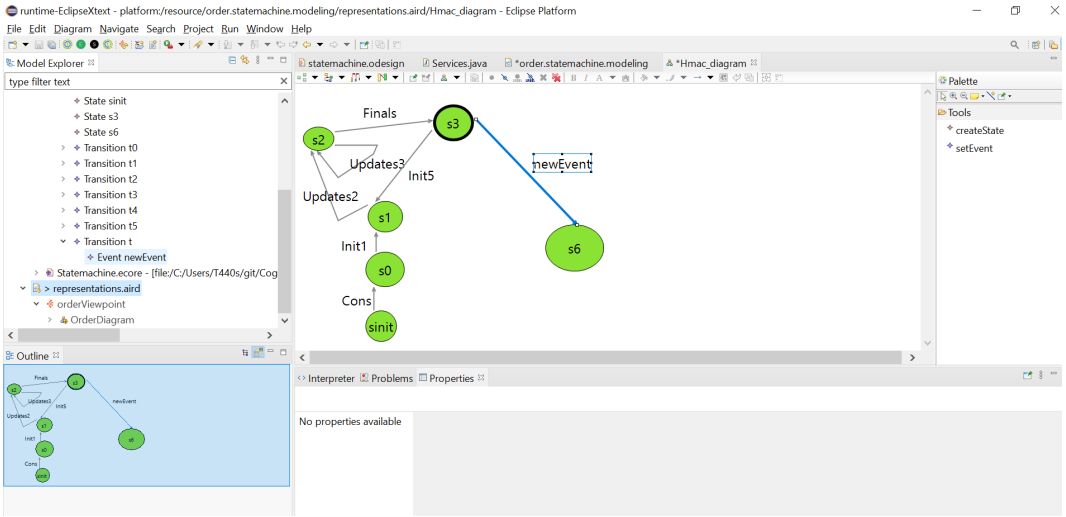
- the nodes can be automatically rearranged by clicking an icon above the diagram, their size can be changed by dragging them, this and more properties can also be adjusted by modifying them in the Properties tab below
- Nodes and edges can be moved by Drag and Drop
- displaying an icon next to the node and edge labels can be set in the Properties tab
- Add states: in the Palette in the right corner, there is an editing tool called “createState”. Click on this, then click into the diagram pane to create a new state.



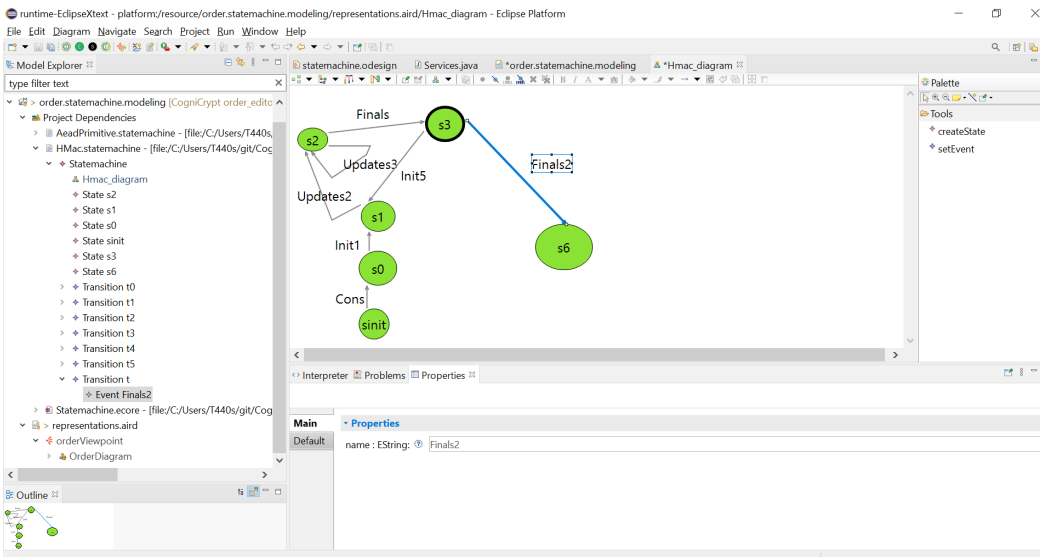
- Add edges: in the Palette click the tool called “setEvent”. Then first click on the outgoing node and then on the target node in the diagram.



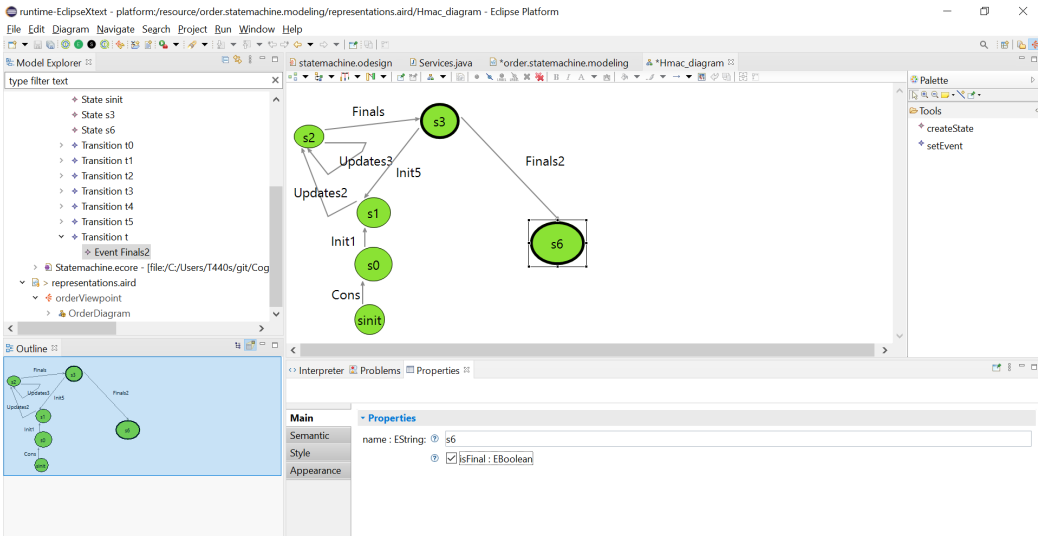
This created an edge between these nodes, setting a default label “newEvent”.



By clicking in the tree editor in the left corner, selecting the current model under “Project Dependencies” and then clicking on the newly generated Transition object, you can change the edge label, i.e. the name of the new Event object: Clicking on the Event opens the responding Properties tab below the diagram and clicking into the name field allows to change the edge label. Press Enter to make the change in the diagram.



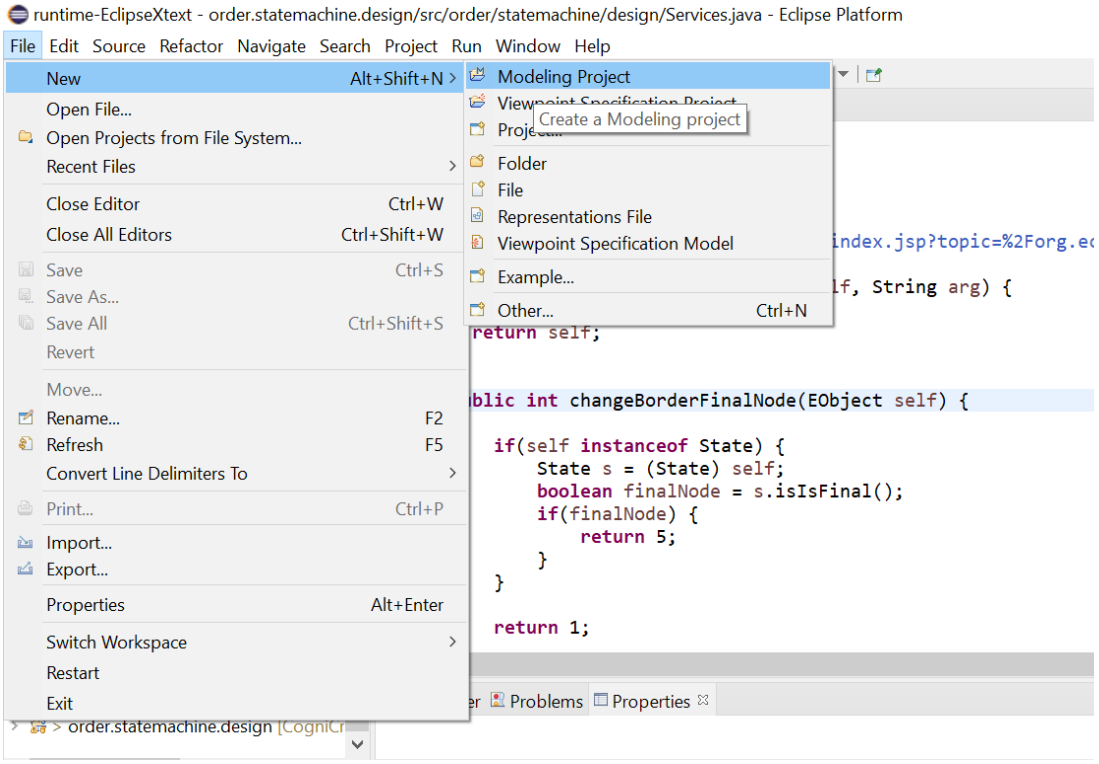
When you click on the new node you can specify in the Properties tab below whether it should be a final node that is displayed with a wider border.

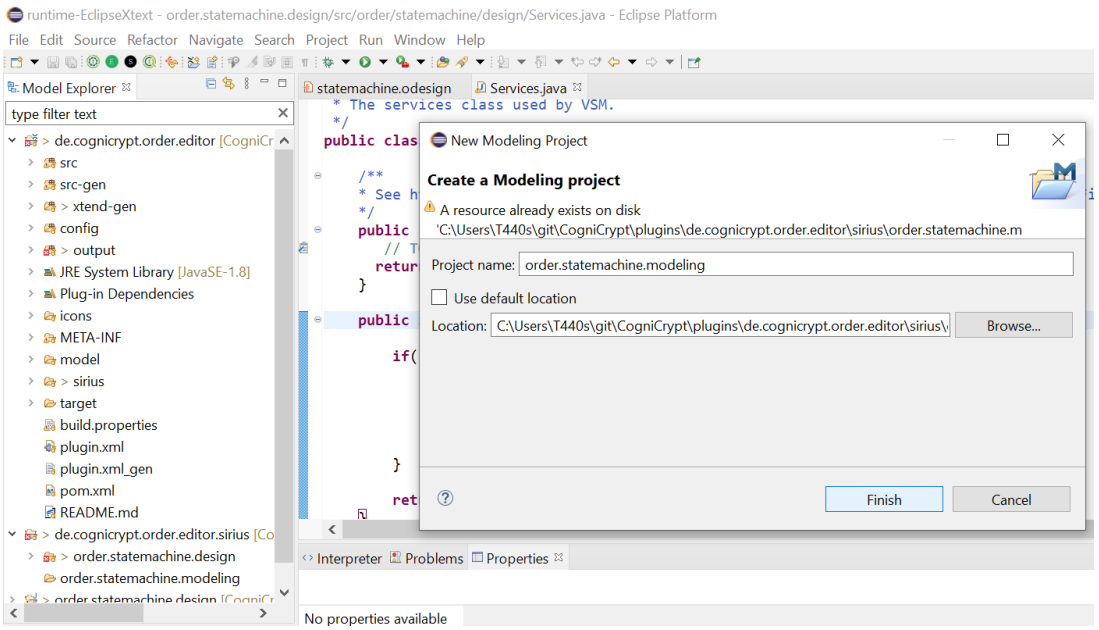


- Remove states: works by pressing the “Del” key and by right-clicking the node, then “Edit” and “Delete from Model”. Connected edges are deleted as well.
- Remove edges: works by pressing "Del" key and by right-clicking edge and selecting “Edit” and “Delete from model”.
- Edit edges: clicking on an edge opens the Properties tab below to change the transition name, style, etc. Changing the edge label, i.e. the Event name, requires to select the current statemachine model in the tree editor on the right (under "Project Dependencies") and selecting the Transition object and then the Event object, which opens the Properties tab below the diagram window to change the edge label name in the name field.
- For more information on Sirius, have a look at their tutorials (https://wiki.eclipse.org/Sirius/Tutorials/StarterTutorial).

Troubleshooting

- If anything does not work as expected, e.g. there are problems adding a statemachine model in the representations view or the diagrams do not display the edges, this can be circumvented by deleting the current modeling project from the workspace and creating a new one in the same location: first delete the representations.aird from the file system, then delete the Modeling project from the workspace, and create a new Modeling project in the same location.





This Modeling project then contains a new and empty representations.aird file. Add

the Statemachine.ecore model from “~\CogniCrypt\plugins\de.cognicrypt.order.editor\model\generated” and repeat the above steps. If there are still problems, also delete all the statemachine models from the output folder and configure them again as described above. Then create the new modeling project.

Missing Features/Issues

- Adjust state names (The name of a new state should be the lowest natural numeral that is not yet a state name). State names are currently possible to edit, though it does not have to be done as there is no meaning in doing so. However, there is no explicit Direct Edit Label tool to entice the developer to edit the state name.
- Constraint that unconnected nodes should not be retained when closing the editor: need to edit the saving mechanism then. Currently saving of the diagram automatically updates its EMF statemachine model and thus also saves the new state, though unconnected.
- More editing tools (e.g. reconnect edge tool, style customization, constraints on deletion)
- Transforming graphical edit into crysl rule edit (“Modifications that do influence the usage pattern and therefore the ORDER expression in the rule should be applied to the underlying CrySL rule whenever the file is saved.”)
- Currently editing a diagram by deleting nodes and edges may invalidate the statemachine model which leads to errors, probably with dependencies. A NullPointerException might occur. Here, it currently needs to be circumvented by deleting the representations.aird as described above and starting everything from scratch, i.e. also delete the statemachine models from the output folder, start a new runtime and create the models again. Then create the new modeling project.
- Proper saving mechanism (currently, it leads to errors when editing and saving a diagram which automatically changes the statmachine model and thus invalidates the model, closing and reopening the editor leads to representation problems for the diagrams, currently need to delete all models and start up everything from scratch)
- Single rule selection