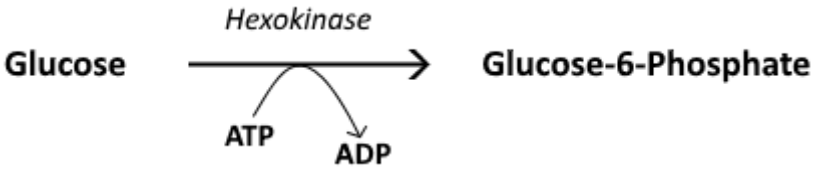
JFX PetriNet: Modelling Enzyme Activity

To illustrate the editor's functionality, we want to model the activity of an enzyme that is part of the glycolysis. The chosen enzyme is Hexokinase that in most organisms phosphorylates glucose to glucose-6-phosphate. The induced reaction can be written as follows:



Biochemical pathways are best modelled using continuous Petri nets. Therefore, the products and substrates will be modelled as continuous places and the enzyme will be modelled as a continuous transition.

Running the Petri net editor:

1. Run the application by double-clicking the "**JFX_PetriNet.jar**" file. Make sure that Java is installed in version 1.8 or higher on your machine.

2. Wait for the application to initialize. You can minimize or close the "Log" window, but keep the "Editor" window open.

Creating a new model:

3. In the "Editor" window, click the "+" section in the top left corner next to the "File" menu (see ***Figure 1***). This will create a new model.

4. Assign the model a name and an author in the "Model" panel on the left side of the "Editor" window. The panels are only visible when a model is active and can be minimized or expanded using the arrows next to the panel titles.

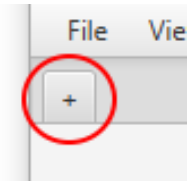


Figure 1: Button for creating a new model.

Creating the places:

5. In the “Tools” panel on the left side click the “Create” button and select “Place” to create an element of this type (see ***Figure 2***).

6. Click anywhere in the white center area of the window to generate a place in the model. A green circle should appear that represents the generated place.

   Notice*: Elements can always be deleted. Select them using the selection frame (left-click and drag the mouse) or by left-clicking an element. Hold “CTRL” to select multiple elements.*

7. Select the generated place (single left-click, don’t double-click). The “Identifier” and the “Properties” panel should appear on the right side of the window.

   Notice*: Double-clicking an element will open the inspector view. You can simply return to the graph view by using the menu bar (“View” -> “View Graph”) or by clicking the orange arrow in the top left corner labelled with “Graph”.*

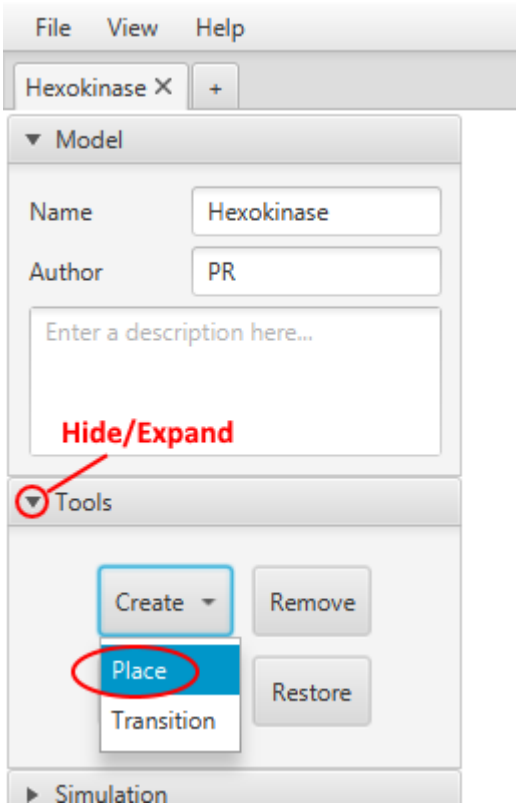


Figure 2: The “Model” and “Tools” panels.

8. In the “Identifier” panel, assign the place the name “Glucose” and an appropriate label such as “Glc”.

9. In the “Properties” panel, make sure that the place is of subtype “CONTINUOUS”. The latest subtype selection will be stored for all newly created elements of the same type. Also make sure that the “Constant” box is NOT checked.

10. In the “Properties” panel, assign the place a token count of 0.57. The minimum and maximum token counts should be left unchanged (default minimum: 0; default maximum: 1.7976931348623157E308).

11. Create three additional places in the same manner using the following properties:

Name	**Label**	**Subtype**	**Constant**	**Token**	**Token (min)**	**Token (max)**
Glucose-6-Phosphate	G6P	*CONTINUOUS*	No	4.2	default	default
ATP	ATP	*CONTINUOUS*	No	2.1	default	default
ADP	ADP	*CONTINUOUS*	No	1.5	default	default

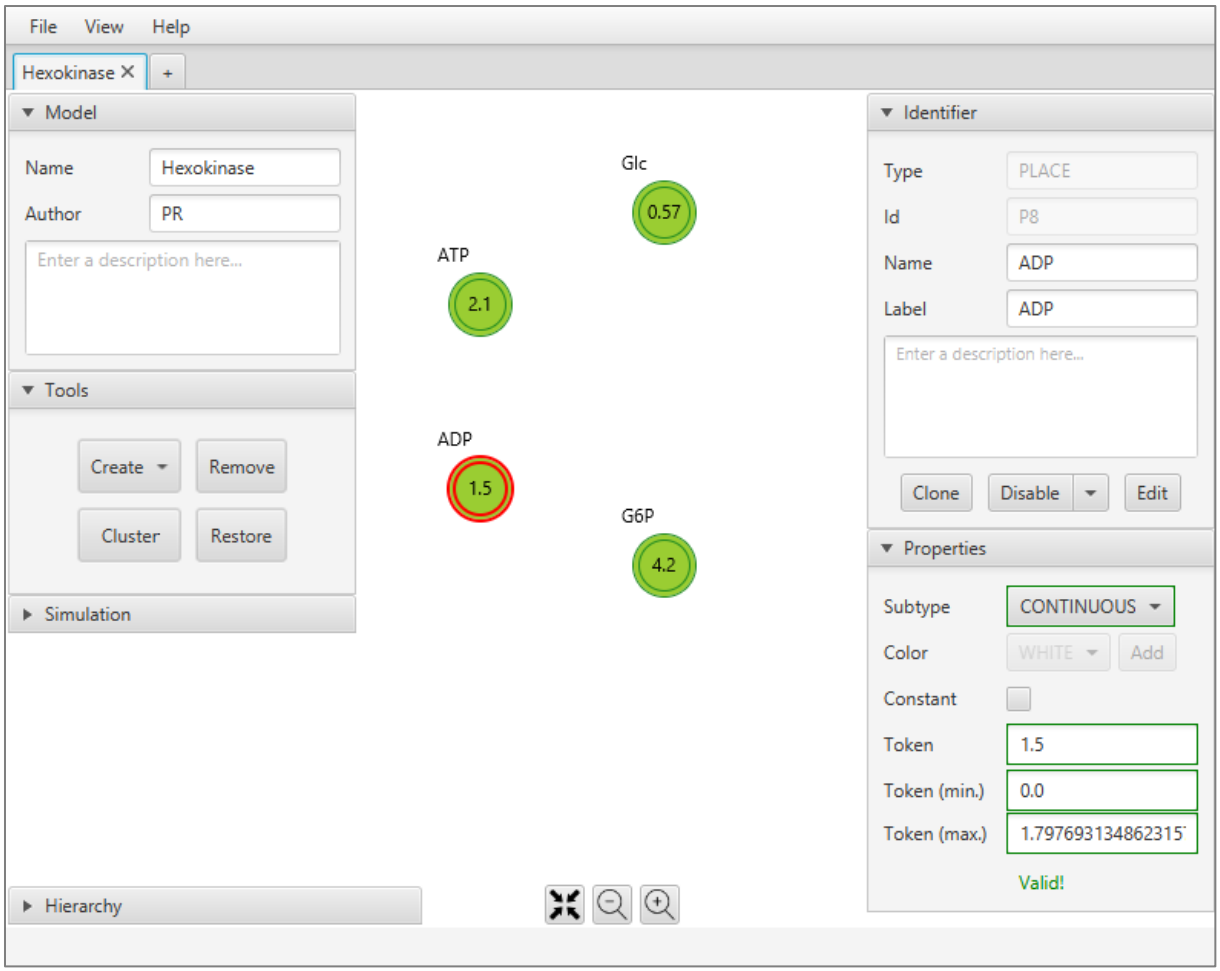


Figure 3: Model state after creating the four places.

Creating the transition:

12. Your model should now look like the model shown in ***Figure 3***. In the "Tools" panel on the left side click the "Create" button and select "Transition" to create an element of this type.

13. Click anywhere in the white center area of the window to generate a transition in the model. A light-blue rectangle should appear that represents the generated transition.

14. Select the generated transition. This time we want to use the inspector view, therefore, double-click the transition or use the menu bar and select "View" -> "View Element Inspector". In the inspector view, make sure you have selected the transition. You can switch between elements by double-clicking the appropriate element in the list on the left side labelled as "Nodes".

15. In the "Identifier" panel in the top middle, assign the transition the name "Hexokinase" and a label such as "HK". Also make sure that the transition is of type "CONTINUOUS".

Creating parameters and assigning enzyme kinetics:

The kinetics of an enzyme usually depend on the concentrations of their substrates and products and various additional parameters. In a Petri net, the enzyme kinetics can be described by assigning speed functions to the respective transitions representing an enzyme. To simplify maintaining these functions, the application supports the use of parameters. These can be created, edited and used in the functions of the transitions presents in a selected model.

16. In the inspector view, use the “Parameter” panel on the right side and create the parameters listed in the following table. The scope determines whether a parameter can be used by other elements. Parameters to create:

Name	Value	Unit	Scope
k_ATP	0.1	mmol	*GLOBAL*
k_Glc	0.0	mmol	*GLOBAL*
k_dGlc	0.37	mmol	*GLOBAL*
v_m	51.7547	mmol/min	*GLOBAL*

17. In the “Properties” panel at the center-bottom left, assign the transition representing the Hexokinase the following (speed) function:

v_m * ATP * Glucose / (k_dGlc * k_ATP + k_Glc * ATP + k_ATP * Glucose + Glucose * ATP)

In the end, your function should look like the function shown below in ***Figure 4***.

Notice*: The values ATP and Glucose depend on the names assigned to the respective places in your model and might be different for your model. Make sure to use the correct names that represent these places, otherwise the function will not be valid (indicated by red borders, otherwise green).*

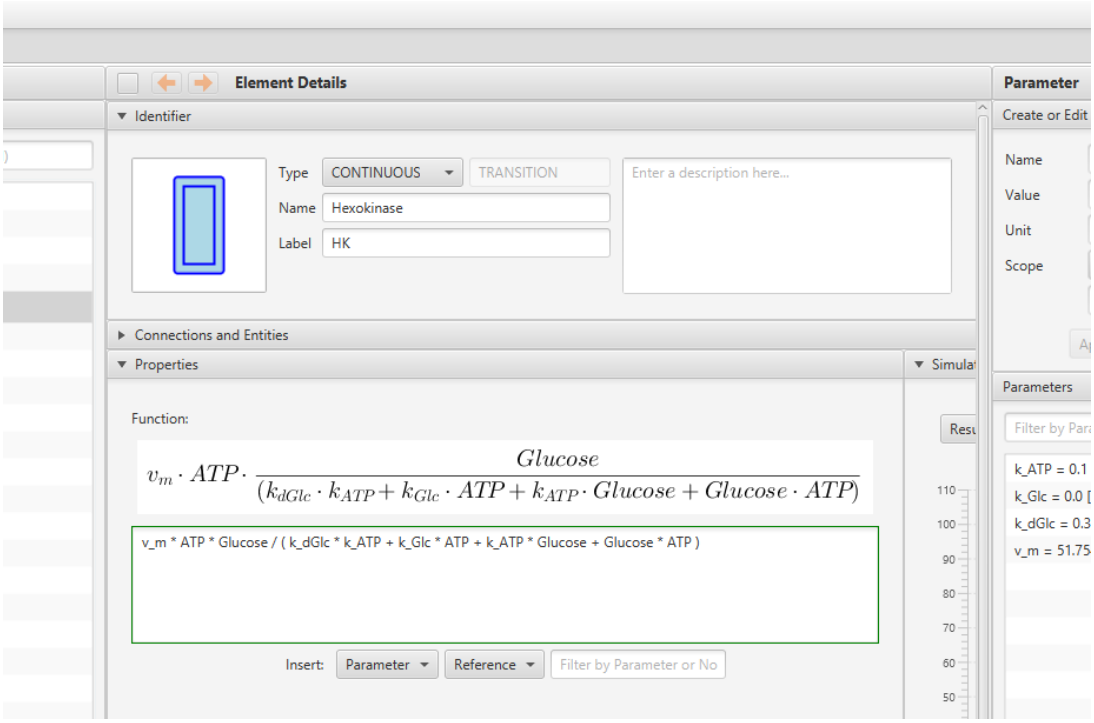


Figure 4: Inspector view showing the (speed) function assigned to the transition "Hexokinase".

Creating arcs/edges:

18. Return to the graph view by clicking the arrow in the top left corner labelled “Graph” or by using the menu bar (“View” -> “View Graph”).

19. Connect the place “Glucose” to the transition “Hexokinase”. Therefore, left-click the place and hold until the place is surrounded by dashed borders. Now drag the mouse over to the transition. You will see an orange arrow following the pointer of your mouse. Release the mouse button on top of the transition. The elements should now be connected by an edge pointing from the place “Glucose” (source) to the transition “Hexokinase” (target).

20. Create three additional edges:
    - from ATP to Hexokinase
    - from Hexokinase to ADP
    - from Hexokinase to Glucose-6-Phosphate

    Once you are finished, your model should look like the model in ***Figure 5***.

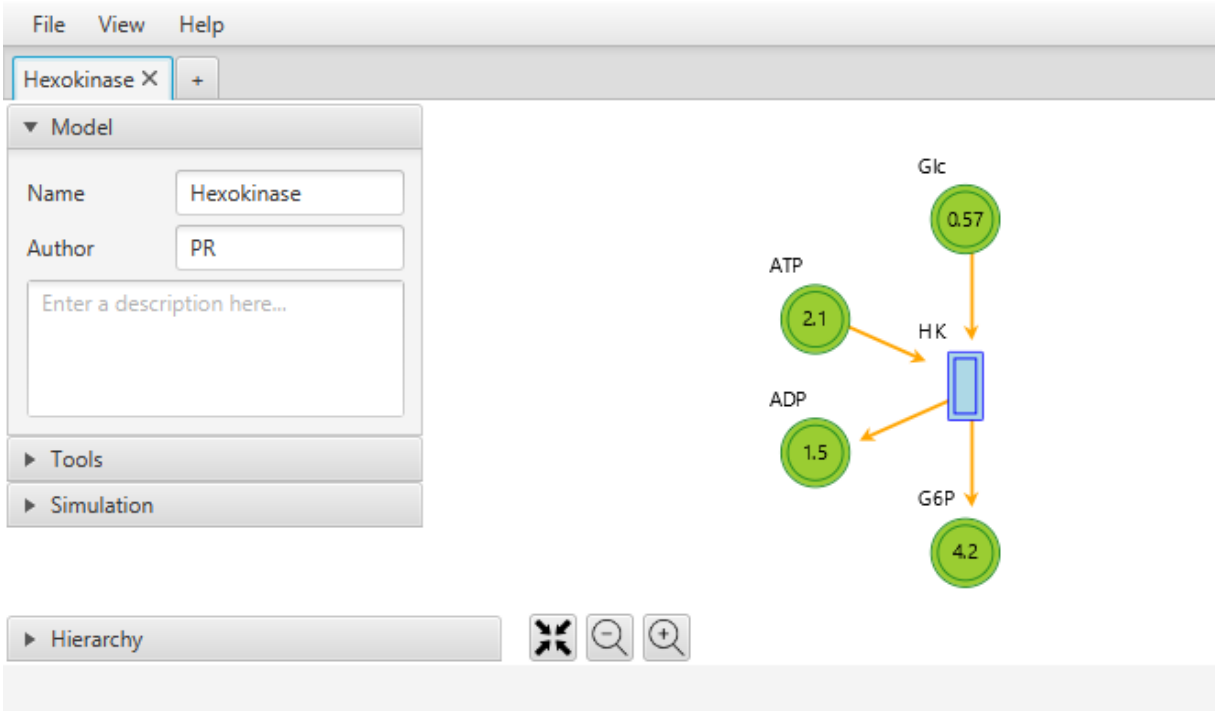


Figure 5: Finished Petri net model of the reaction induced by Hexokinase.

JFX Petri Net: Simulating Complex Metabolic Pathways

The first chapter of this tutorial introduced you to the basic editing features of the application. We will now proceed by looking at its simulation capabilities that are provided by OpenModelica.

While modelling small components such as the activity of an enzyme can be done fast and easily, modelling an entire biochemical pathway is much more difficult and time consuming. Therefore, we will import an existing model to the application that is a lot more complex than the model generated in the first chapter.

The application allows you to import previously created models from XML and SBML (which is also used by VANESA).

Import and simulate the model:

1. In the main window's menu bar, use the "File" menu and click "Open". Select and open the file "**hynne_reverse.xml**". Upon success, the model will be opened in a new tab. You can try to find the components we modelled in the first chapter within the imported model.

2. In the graph view, use the "Simulation" panel on the left side to specify a **Stop Time** of 2 and 300 **Intervals**.

3. Open a new results viewer using the appropriate button in the simulation panel. A new window opens that can be used to display results for any already performed and all upcoming simulations.

4. In the "Simulation" panel in the main window, press "Start". The application will export the model, compile and build an executable simulation using the OpenModelica compiler, and will then run the simulation. The results viewer window will print a notification that new simulation results are available, even while the simulation is still in progress.

   ***Notice**: The simulation process will fail to run if OpenModelica is not installed on your local machine. But not to worry. Instead you can import prepared simulation results in the results viewer. In the results viewer, use "File" > "Import" and select the "**hynne_reverse_results_(2-300).xml**".*

5. Use the results viewer window and show the results for a few places and/or transitions (i.e. ATP, ADP, Glucose, Glucose-6-Phosphate, Hexokinase). If the simulation is still running, you need to refresh the results by removing/adding them to the chart. You can also open multiple results viewer windows at once to inspect multiple data sets in parallel.

Try to interpret the generated results. What data are the charts showing? What is the time frame that has been simulated (unit)?