**A Tutorial for Physiologically Based Toxicokinetic (PBTK) Modeling in Petri Net (PN)**

# Install Snoopy

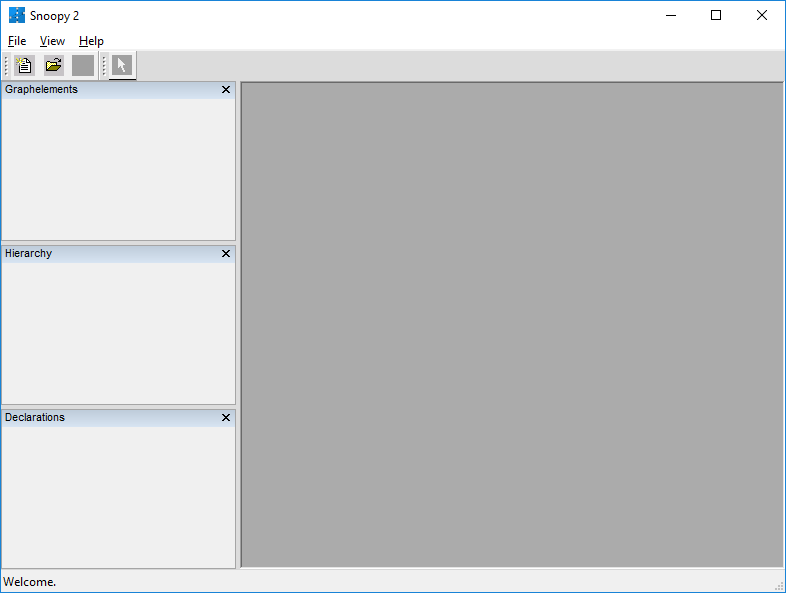
Snoopy is available as a package for Windows, Linux, or Mac OS X. The screenshots used throughout this tutorial were taken from a Windows installation.

## Locate and download the appropriate version of Snoopy at <http://www-dssz.informatik.tu-cottbus.de/DSSZ/Software/Snoopy#downloads>

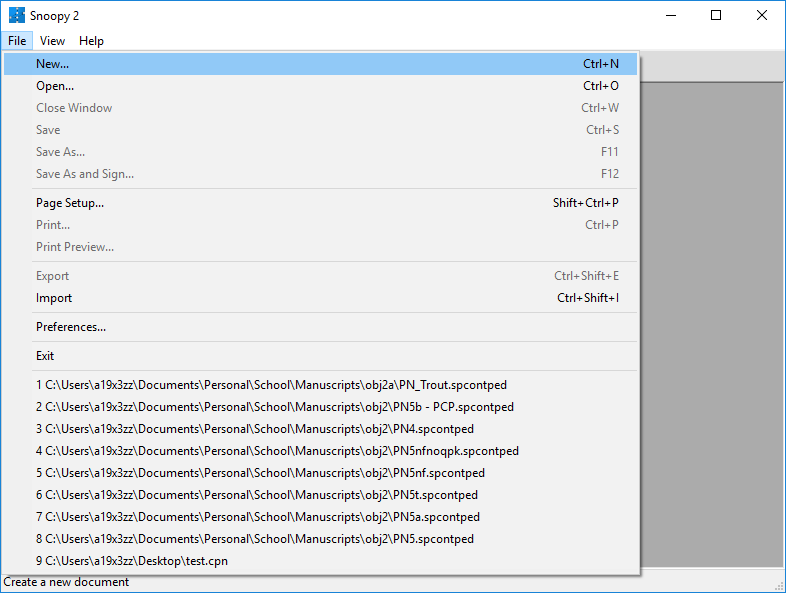
## Install according to the instructions on the website

# Create File

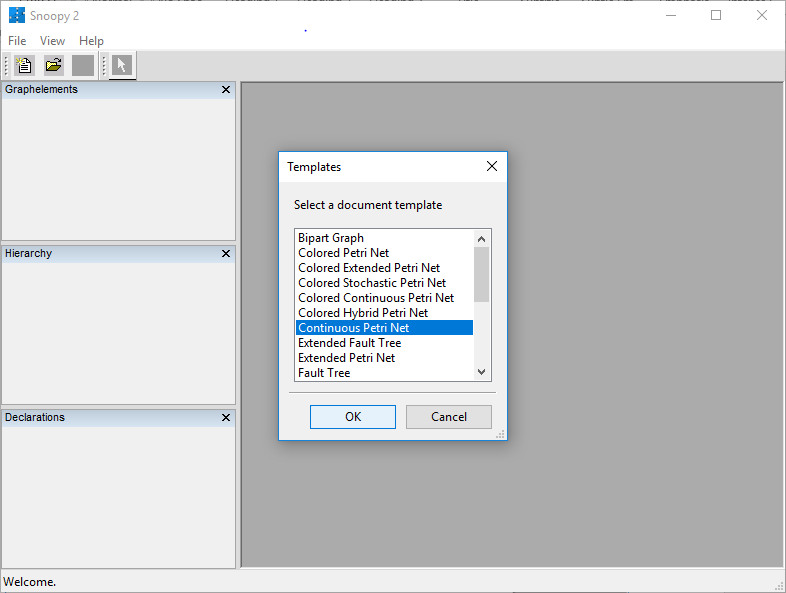
## Open the Snoopy application



## Click on ‘File’ -> ‘New’



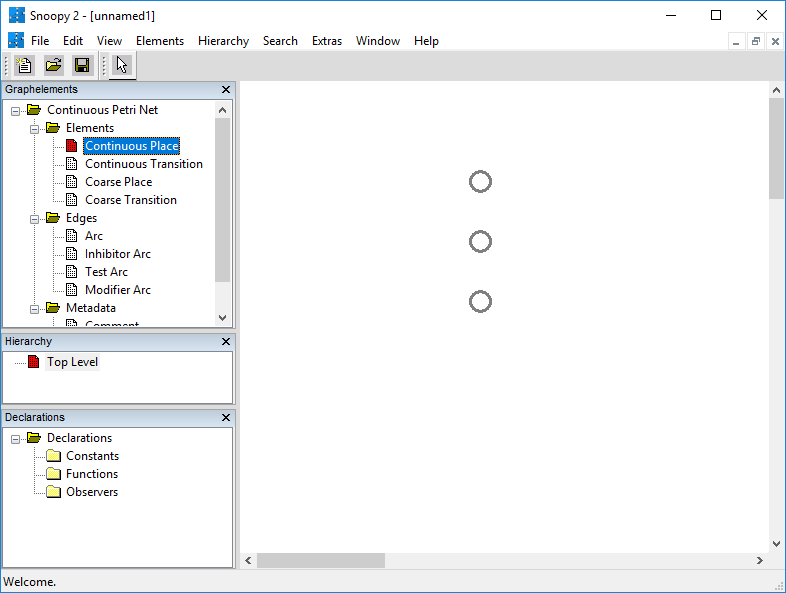
## Click on ‘Continuous Petri Net’, then ‘OK’



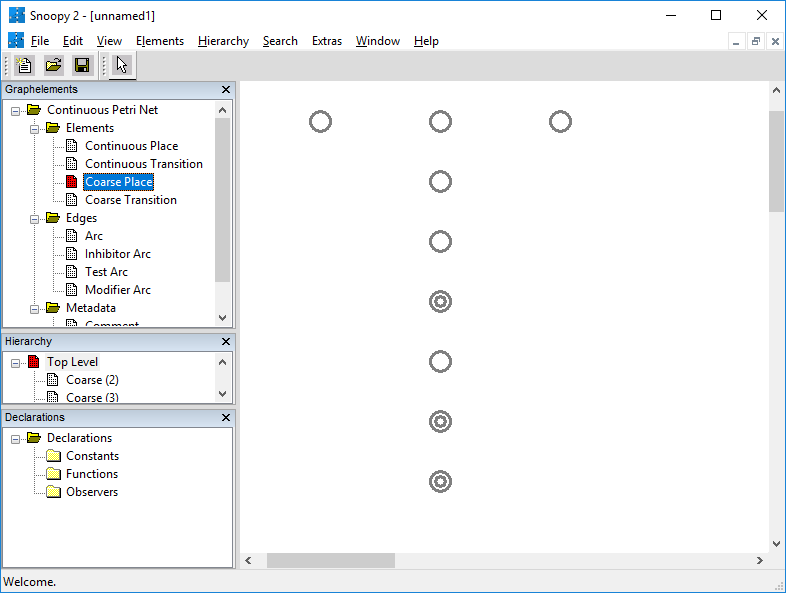
# Create Places

Places are created to store ‘Tokens’, which we use to represent the toxic compound whose flow is being modeled. Create ‘Continuous Places’ for each of the non-metabolizing compartments, arterial blood, and venous blood. Create ‘Coarse Places’ for the metabolizing compartments. Arrange the compartments in a vertical line down the center, with venous blood to the left and arterial blood to the right.

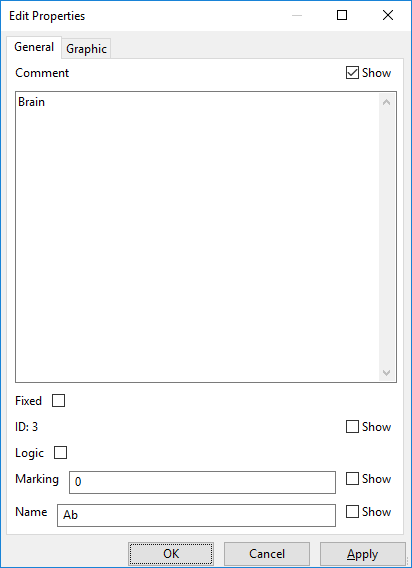
## Click on ‘Continuous Place’ in ‘Graphelements’, then click on the drawing pane where you want the place positioned. Each click on the drawing window will create a new place. You may click on the selector tool (arrow) to reposition a place.



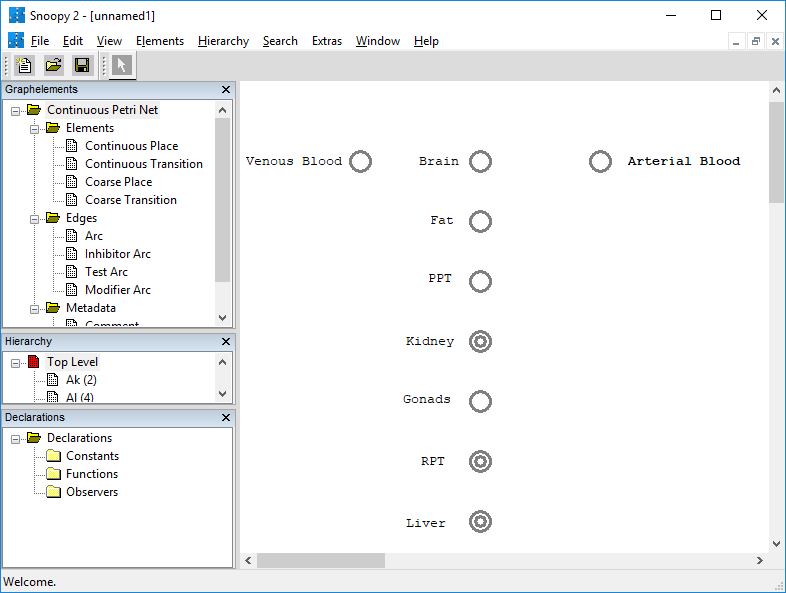
## Click on ‘Coarse Place’ to place the coarse places in the same manner



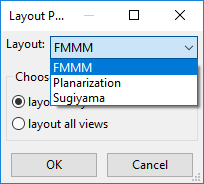
## Double-click on each of the places to edit the properties. Enter a description in the ‘Comment’ area and ‘Show’ it on the net. Use the abbreviation in the ‘Name’ box, which will be used later. Do not ‘Show’ the ‘Marking’ or ‘Name’. Leave the initial ‘Marking’ set at ‘0’.



## You may click and drag the descriptions on the drawing pane as needed to make the PN more readable.



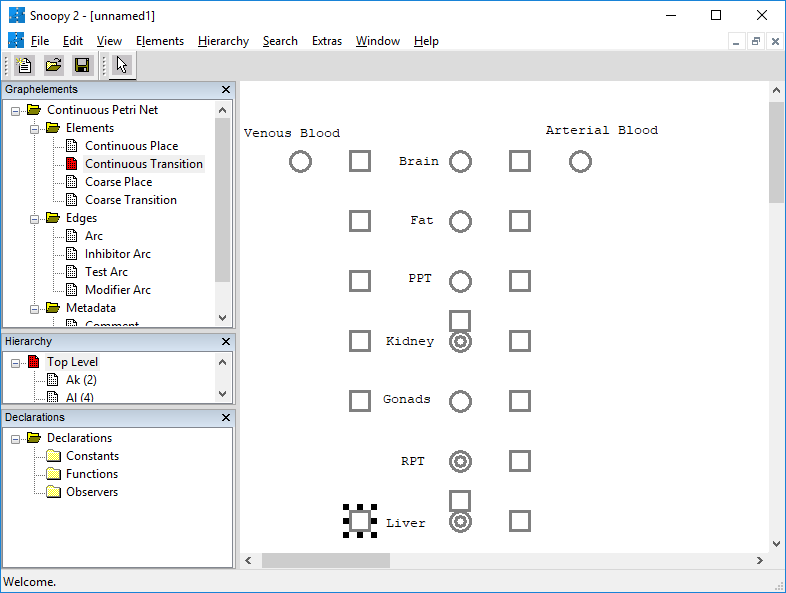
## There is a ‘Layout’ window available under ‘Edit’ -> ‘Layout’ to rearrange places and transitions, but I have not found it to be useful.



# Create Transitions

Transitions are created wherever there is movement of tokens from one place to another. Create ‘Continuous Transitions’ between the arterial blood and each of the compartments, each of the compartments and venous blood, and wherever there is flow between compartments (ie. PPT to Kidney and RPT to Liver). In this model, which assumes that all blood flow moves through the liver, there is an exception for RPT which lacks a transition to the venous blood. Create a ‘Coarse Transition’ between the venous and arterial blood, which will be used to determine recirculation and gill flux.

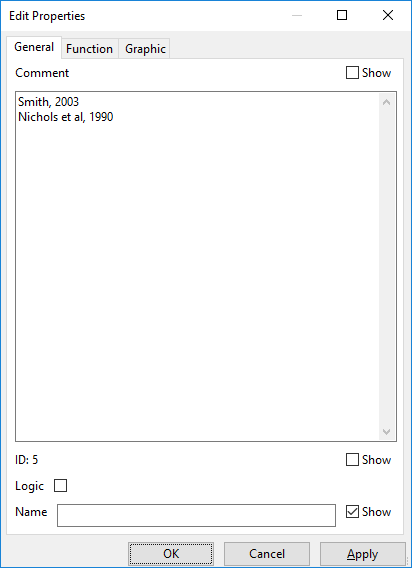
## Click on ‘Continuous Transition’ in ‘Graphelements’, then click on the drawing pane where you want the transition positioned. Each click on the drawing window will create a new transition. You may click on the selector tool (arrow) to reposition a transition.



## Click on ‘Coarse Transition’ to place the Gill\_Flux transition in the same manner.



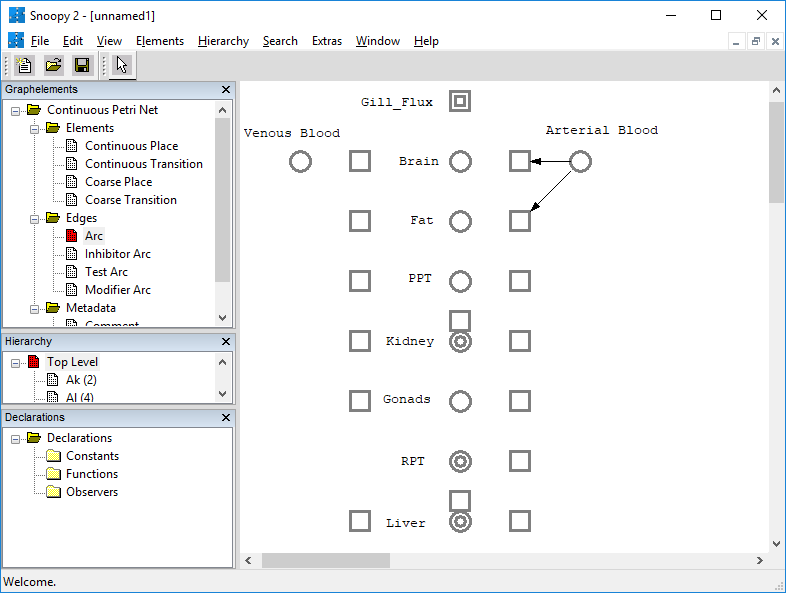
## Double-click on each of the transitions to edit the properties. I entered and showed the ‘Name’ for ‘Gill\_Flux’, but not for the other transitions. In the ‘Comments’ section, I find it useful to cite the literature where I derived the equations from, without ‘Show’ing it on the screen.



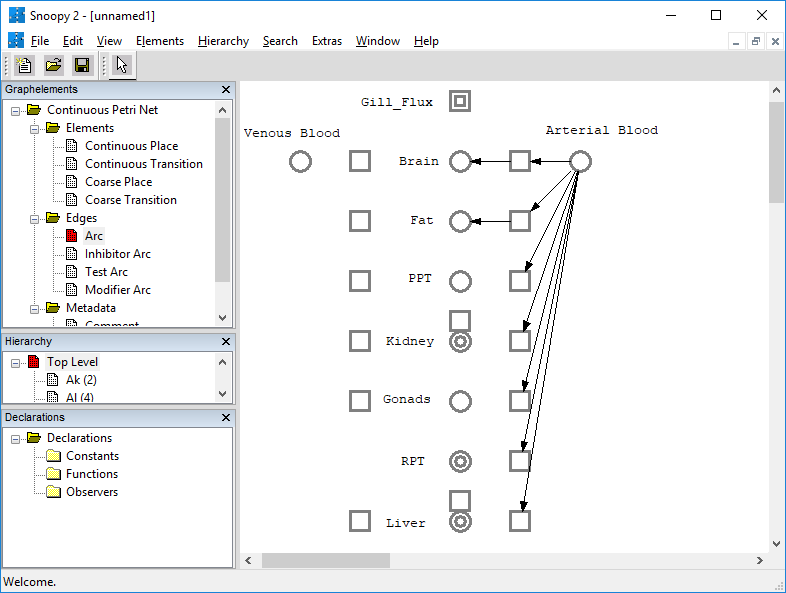
# Create Arcs

Arcs are used to define how tokens move through the net. Arcs are positioned from a place to a transition, or from a transition to a place; they are never used between two places or two transitions. Directionality of the arc is critical for token flow.

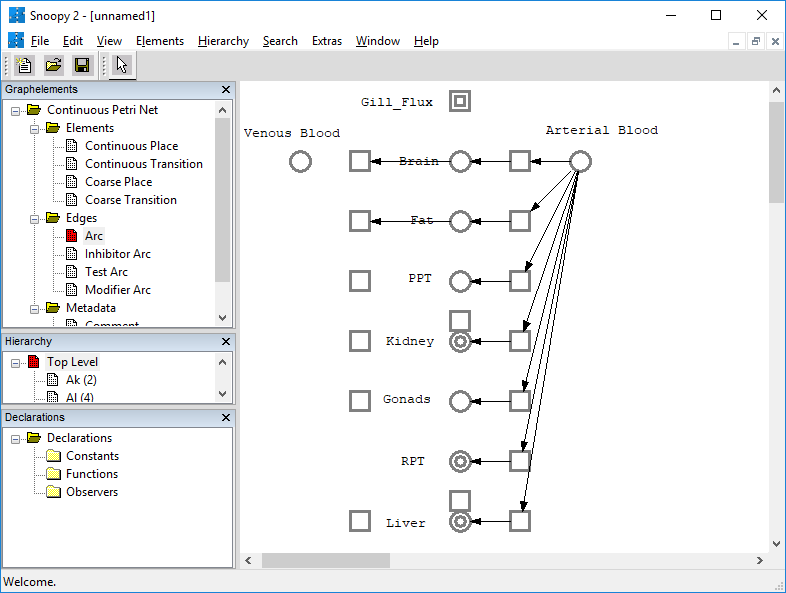
## Draw arcs from the arterial blood to each of the inbound transitions for a compartment. Click on ‘Arc’ in ‘Graphelements’, then click and drag from the arterial blood place to a transition. Each click on the drawing window will create a new arc.



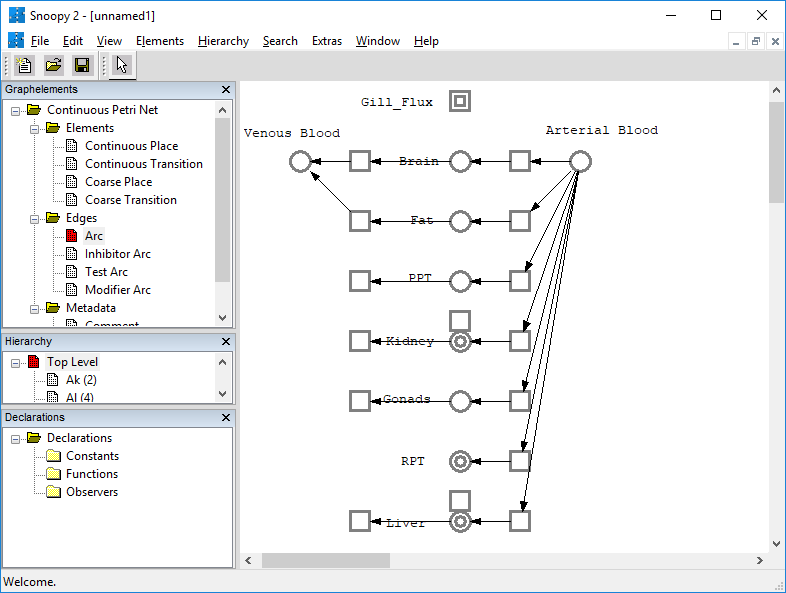
## Draw arcs from each of the inbound transitions to its compartment.



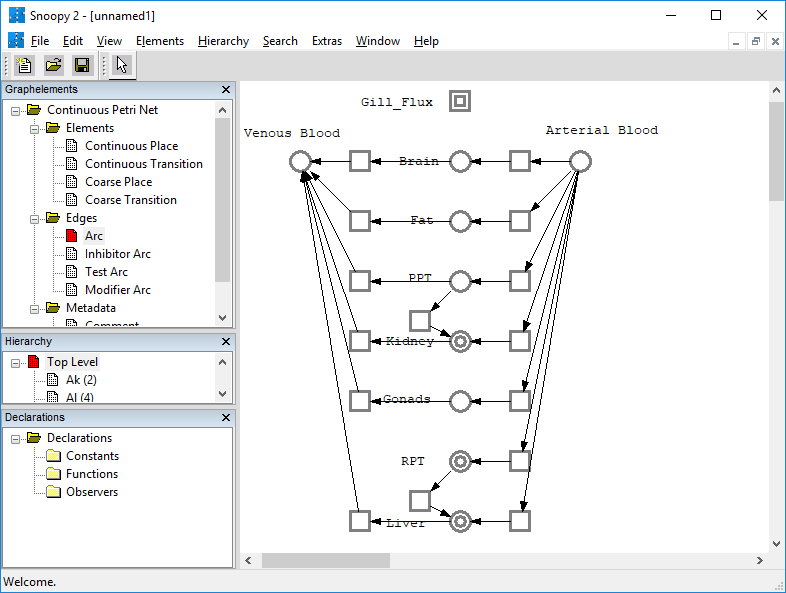
## Draw arcs from each compartment to its outbound transition.



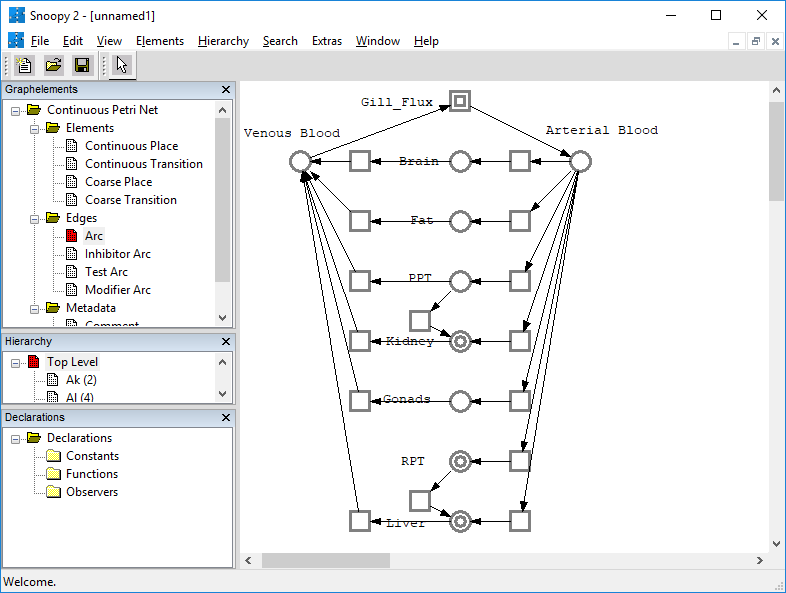
## Draw arcs from each outbound transition to the venous blood.



## Draw arcs directing token flow from the PPT -> transition -> Kidney and from the RPT -> transition -> Liver.



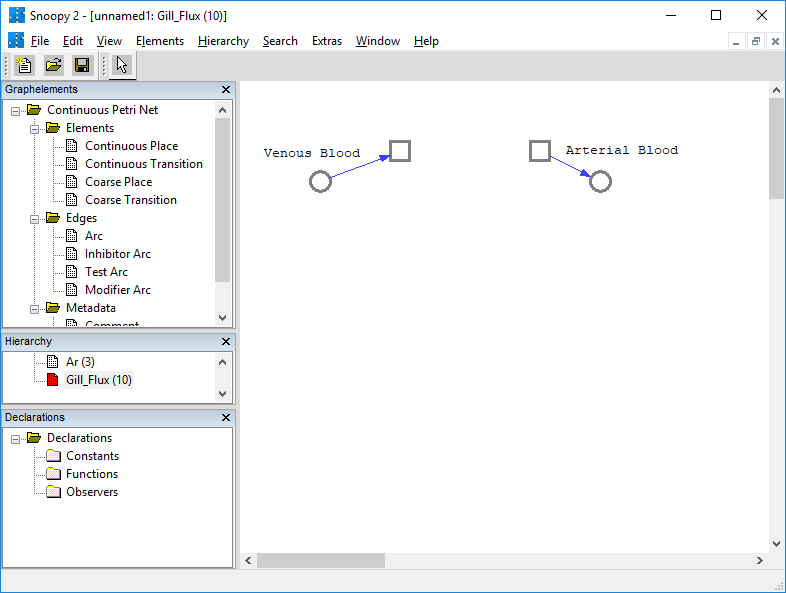
## Draw arcs directing flow from the venous blood, through the gills, and into the arterial blood.



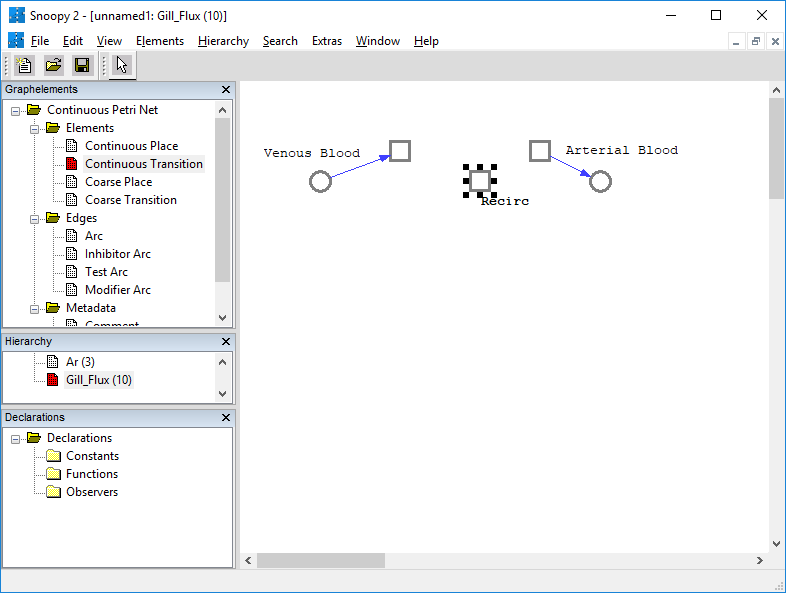
# Gill\_Flux Coarse Transition

The coarse transitions create a subnetwork to group additional places and transitions, without overloading the top-level network.

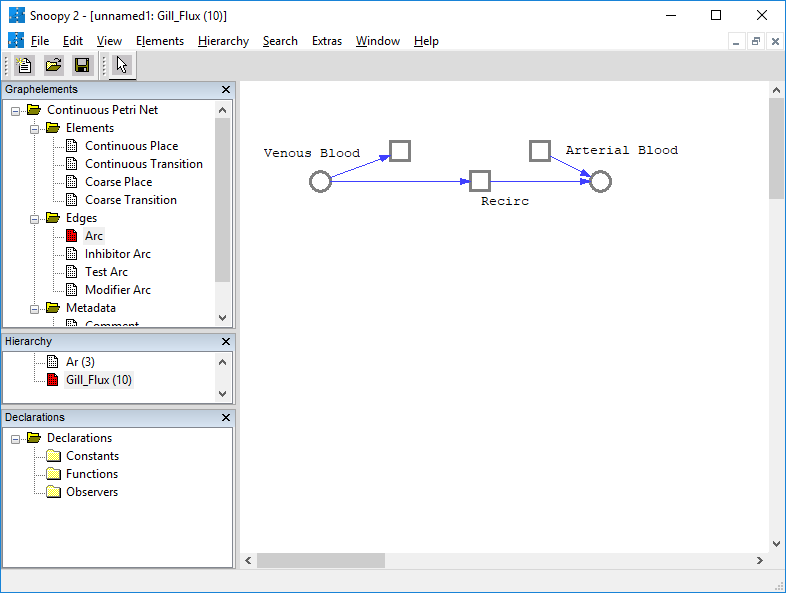
## Double-click on the ‘Gill\_Flux’ subnetwork in the ‘Hierarchy’ menu. Transitions have automatically been created from the venous blood and to the arterial blood. We will use these transitions to model the compound being expelled from the venous blood into the water and inspired from the water into the arterial blood.



## Create an additional continuous transition called ‘Recirc’ between the venous blood and arterial blood. We will use this to show recirculation of tokens.



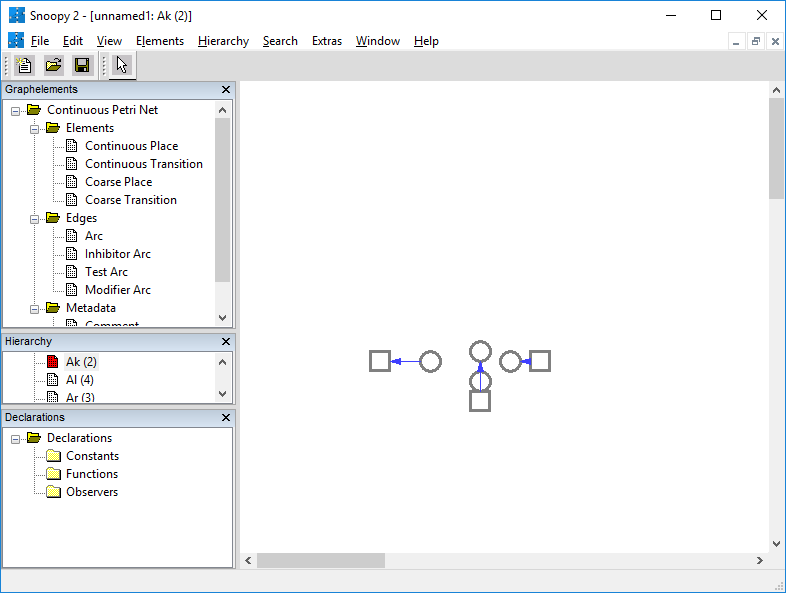
## Connect ‘Recirc’ from the venous blood to arterial blood with arcs.



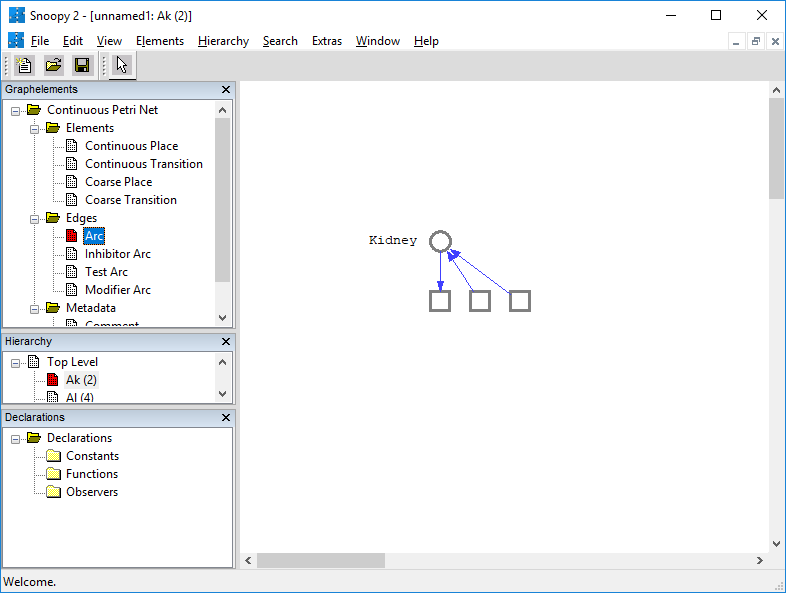
# Metabolizing Coarse Places

Like coarse transitions, coarse places create a subnetwork to group additional places and transitions, without overloading the top-level network. Below, the kidney compartment is shown, but other metabolizing compartments should be treated similarly.

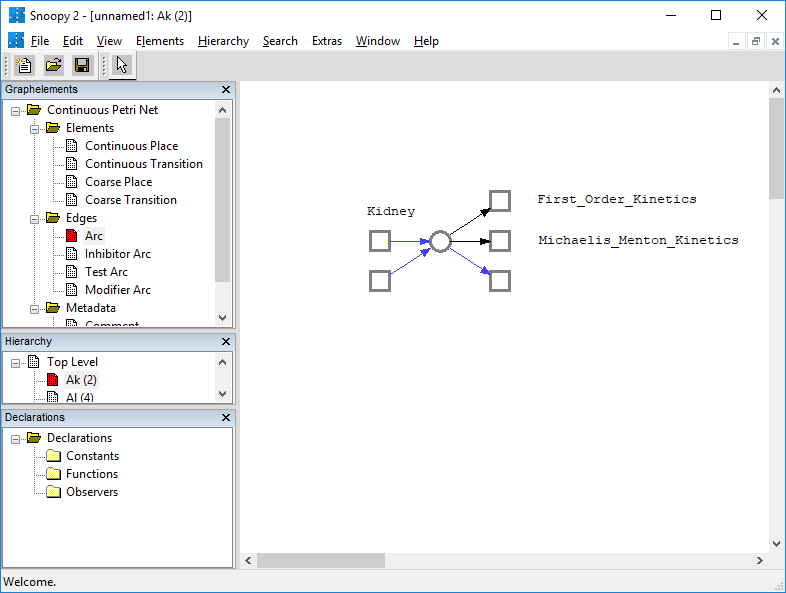
## For each of the metabolizing compartments, double click on its subnetwork. Places, transitions, and arcs have automatically been created for each of the coarse places to signify the token flow moving in and out of that place. For instance, in the kidney coarse place, there are three transitions and three places. Two transitions are directed toward a place with arcs (inbound flow), and the third place is directed toward a transition (outbound flow).



## For our purposes, we only need one place (ie. kidney). Delete two of the places and redraw arcs from the associated transitions, preserving directionality.



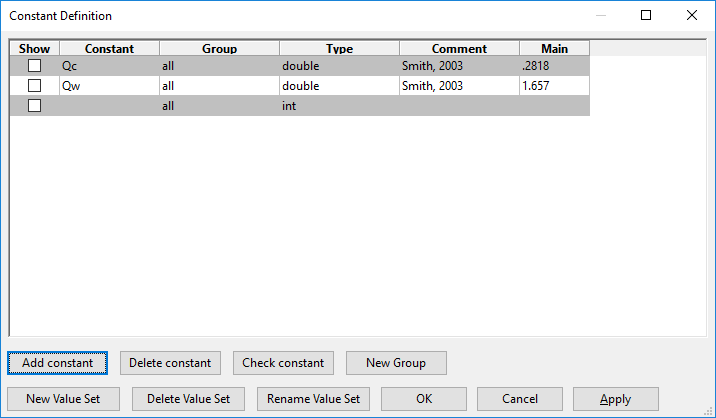
## Add additional continuous transitions and arcs for metabolism.



# Constant Definition

The physiological, physiochemical, and other parameters required for the mathematical calculations are declared as constants. Further information about each of the parameters is available in the accompanying manuscript.

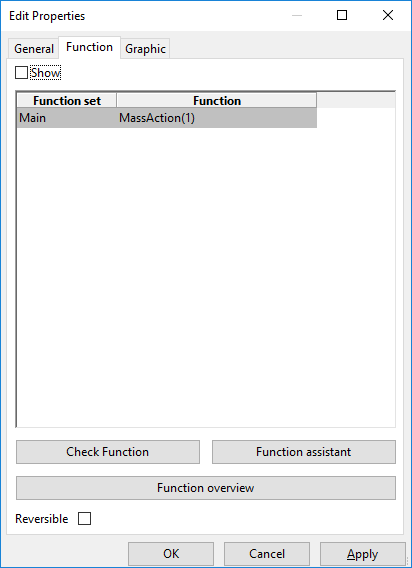
## Double-click on the ‘Constants’ folder under the ‘Declarations’ sidebar. Click on the ‘Add constant’ button. Fill in the fields for each parameter. ‘Constant’ should be the abbreviation of the parameter. ‘Group’ should be left as ‘all’. ‘Type’ should normally be set as a ‘double’. Under the ‘Comment’ field, it might be useful to cite where you obtained this parameter. Under ‘Main’, the value of the parameter should be entered. Continue to ‘Add constants’ until the table contains all of the required parameters.



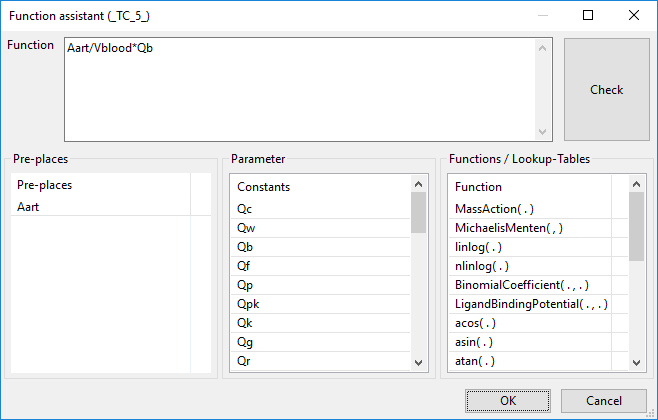
# Function Definition

The mathematical equations that are used to simulate the model are contained within the transitions. The equations are referred to as ‘Functions’ in Snoopy and set in the ‘Function’ tab within the transition properties.

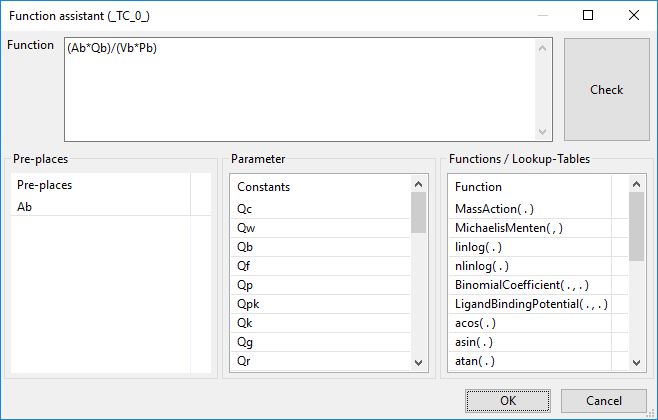
## Define transitions from arterial blood to compartments. Double-click on each transition. Click on the ‘Function’ tab.



## Click on the ‘Function assistant’ button. The available pre-places (places connected to the transition with an inbound arc) are listed, as well as the parameters that were defined in the constant declaration. The function for the transition between arterial blood to a specific compartment (ie. brain) is straightforward. Arterial concentration is multiplied by the blood flow to the compartment (Qb). The pre-place, Aart, is a mass unit, so it is divided by blood volume (Vblood) to calculate concentration. The formula can be entered into the function assistant as shown below. Pre-places or parameters can be double-clicked to reduce typing and prevent typographical errors.



## Next, define the outbound transitions from each compartment to the venous blood (excluding RPT, as discussed above). For most of the transitions, the formula is the concentration of chemical in a compartment (Qb/Vb for the brain) multiplied by the blood flow to the compartment (Qb) and divided by the tissue/blood partitioning coefficient for the tissue (Pb). The formula can be entered into the function assistant for a specific transition as shown below.



## In compartments where a portion of the blood flow is routed to another compartment rather than the entirety of blood flow to the venous blood, then the partial flow needs to be taken into account. For instance, the transition between the kidney and venous blood is written as equation 1, where Qpk is the fraction of PPT blood flow which goes to the kidney rather than venous blood.

(1)

## Similarly, the transition from PPT to the kidneys is written as equation 2.

(2)

## Since the model assumes all the blood exiting the RPT goes to the liver, the transition from the RPT to liver is the same as it would be if all the blood flowed to the venous blood compartment, shown in equation 3.

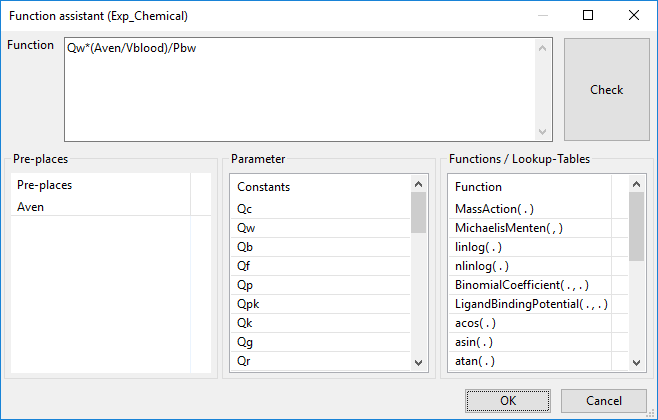
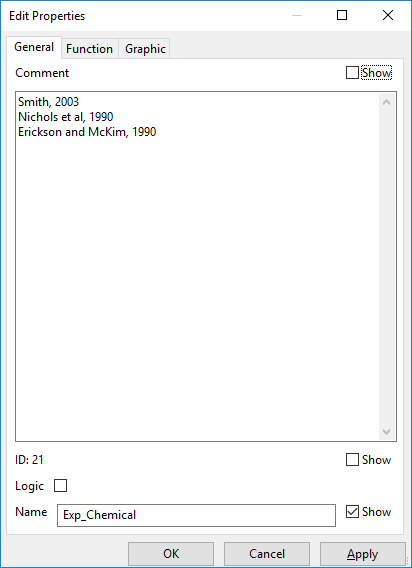
(3)

## The transition exiting the liver must include the additional blood flow from the RPT, so is written as shown in equation 4.

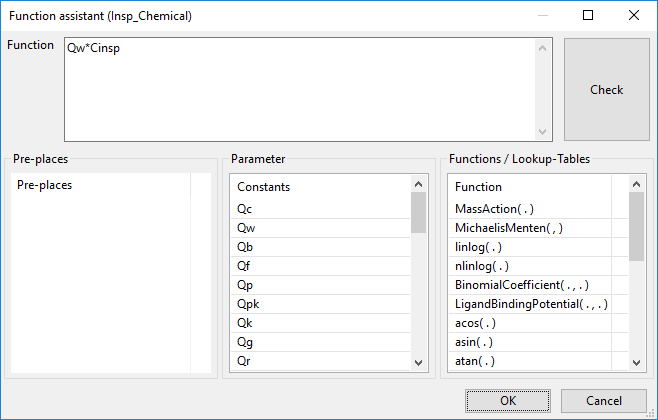
(4)

## The functions within the ‘Gill\_Flux’ subnetwork must be defined. The open-ended transition exiting the venous blood compartment is labeled ‘Exp\_Chemical’, signifying the chemical expired from the gills. The open-ended transition into the arterial blood compartment is labeled ‘Insp\_Chemical’, and the transition linking the venous blood to the arterial blood is labeled ‘Recirc’ for recirculation.

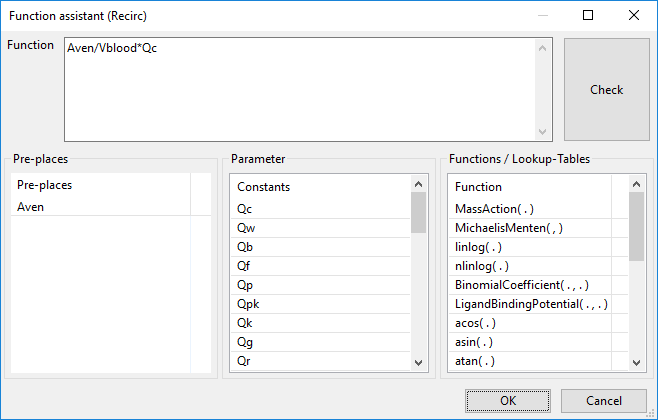
## The properties and function for the ‘Exp\_Chemical’ transition are shown below.



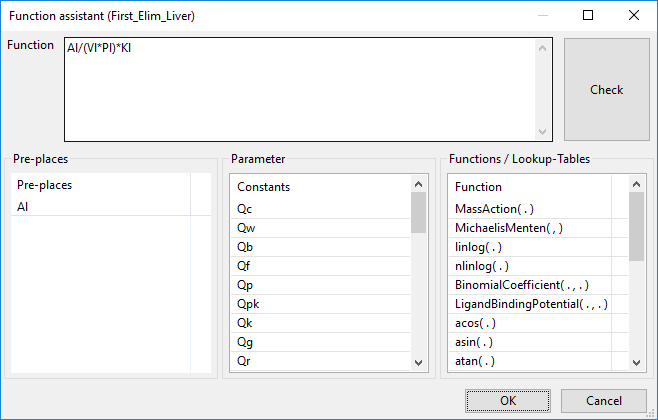
## The function for the ‘Insp\_Chemical’ transition is entered as follows, where ‘Cinsp’ is the concentration of chemical in the water.



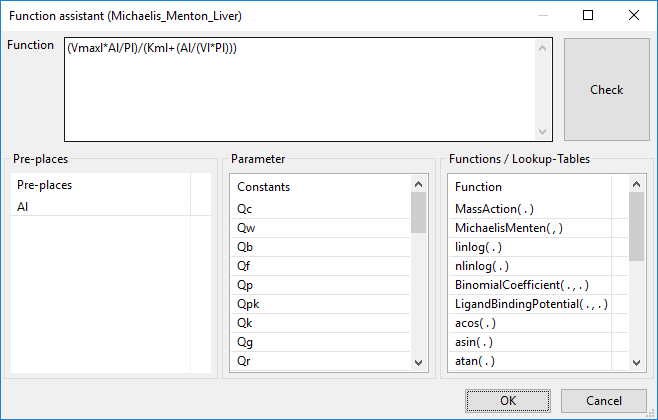
## The ‘Recirc’ transition accounts for the chemical in the venous blood returning to the arterial blood compartment. The formula is simply the venous blood concentration times the cardiac blood flow (Qc), and is entered as shown below.



## The final transitions to be entered are for the metabolizing compartments. A chemical that is metabolized is considered to exit a compartment, since it is no longer in its original form. Places could be created to capture the metabolites and potentially be used to generate a PBTK model for the metabolites, but that is beyond the scope of this paper. The present model contains both first order and Michaelis-Menton kinetics. For first order kinetics (ie. in the liver), the function is written as shown below, with Kl being the first order constant.



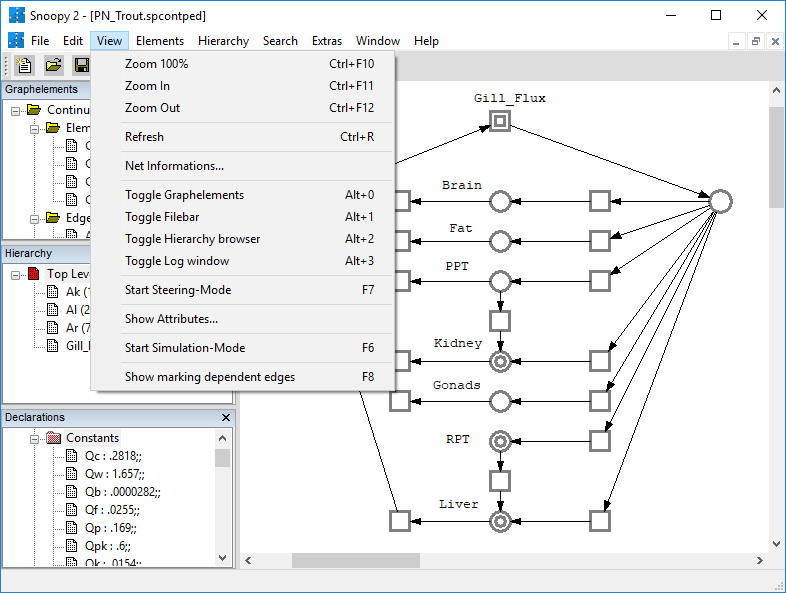
## Michaelis-Menton kinetics are written as shown below. The function utilizes the maximum rate constant (Vmaxl) and Michealis constant (Kml).



# Model Simulation

Once the model is complete with all places, transitions, arcs, and constants defined, it is ready for simulation. The simulation generates a time-course data set for each of the places within the Petri Net.

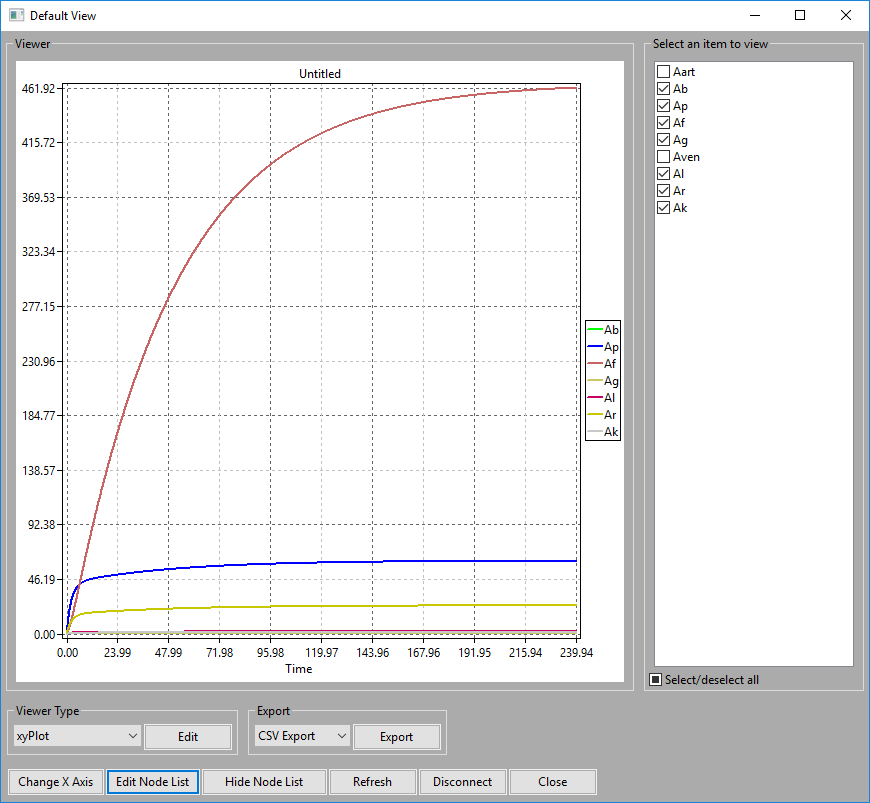
## Simulation-Mode is found in the ‘View’ menu or by pressing the ‘F6’ key.



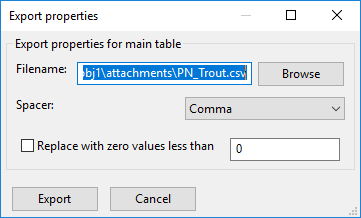
## The values in the ‘Model Configuration’ and ‘Import/Export Details’ are left as defaults. The ‘Simulator Configuration’ menu defines the time-course for the simulation. The present model is simulated for 240 hours, therefore ‘Interval start’ is set to ‘0’ and ‘Interval end’ is set to ‘240’. The MATLAB model, from which this model was derived, generated 3916 time data points, therefore ‘Interval Splitting’ is set to ‘3916’. ‘Simulator Semantics’ and associated properties were left to defaults. The ‘ODE Solver’ and associated properties were also left at default values in the present study, but different solvers and properties could be explored in the future.

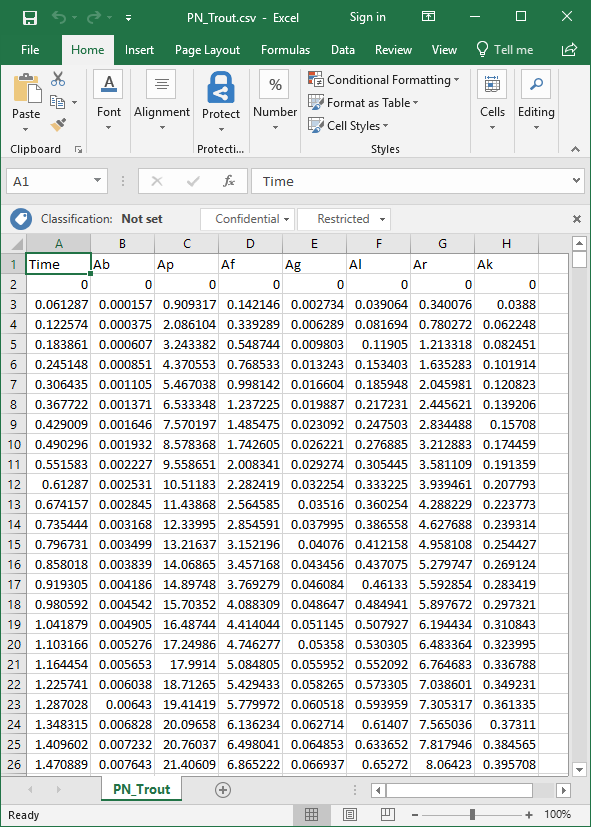
## 

## Double-click on the ‘Default View’ to open the simulation viewer. The node list on the right of the screen should include all places within the Petri Net. The ‘Edit Node List’ button may be used to ensure that all the desired places are available for viewing should places be added after initial simulation. Places may be selected/deselected as desired for graphical viewing. Pressing the ‘Start Simulation’ button will generate the time-course graphs for the selected places. It should be noted that the plots are shown in mass units, not concentration.



## Time-course simulation results may be exported as a comma separated value (.csv) file for use in other spreadsheet programs. Click on the ‘Export’ button to open the ‘Export properties’ screen. Fill in the appropriate filename or browse to the appropriate directory. Click ‘Export’ to save the time-course data for the selected places. The output .csv file is shown below in Microsoft Excel.

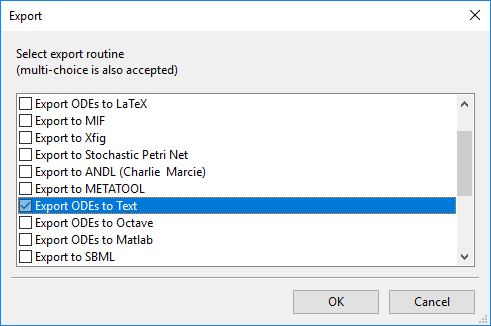




# Model Export

Snoopy has several options for exporting the model.

## A modeler may find it useful to look at the ODEs which are generated from the Petri Net model. This can be done with the File -> Export utility shown below. The output ODEs are also shown. The model can be output to a variety of other formats as well.





## Direct Export is accessed from the ‘Simulation-Mode’ screen under the ‘Import/Export Details’ menu. A parsed ODE text file is generated, which substitutes values for the constants. These ODEs may be used in other solving software. An example parsed ODE is shown below.

