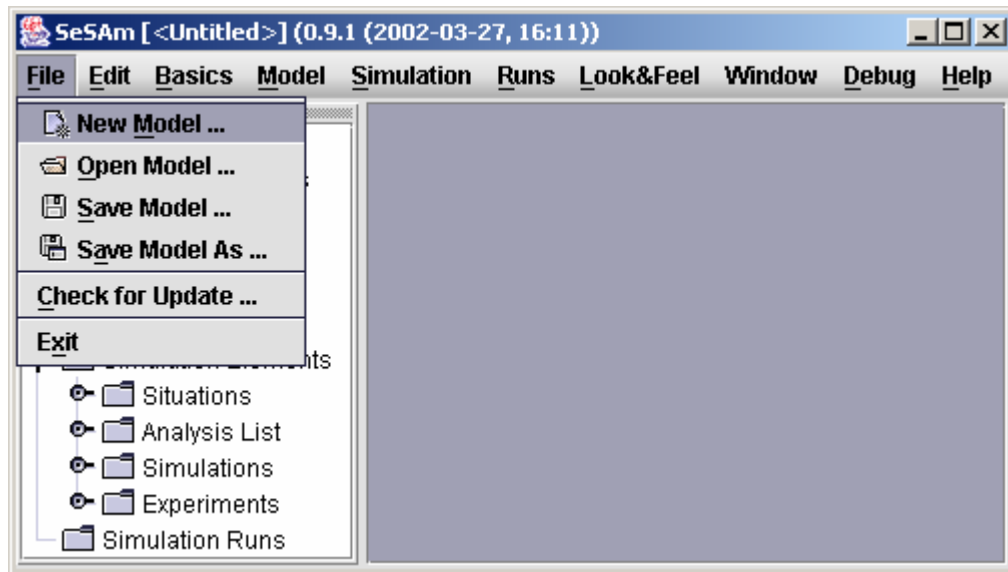
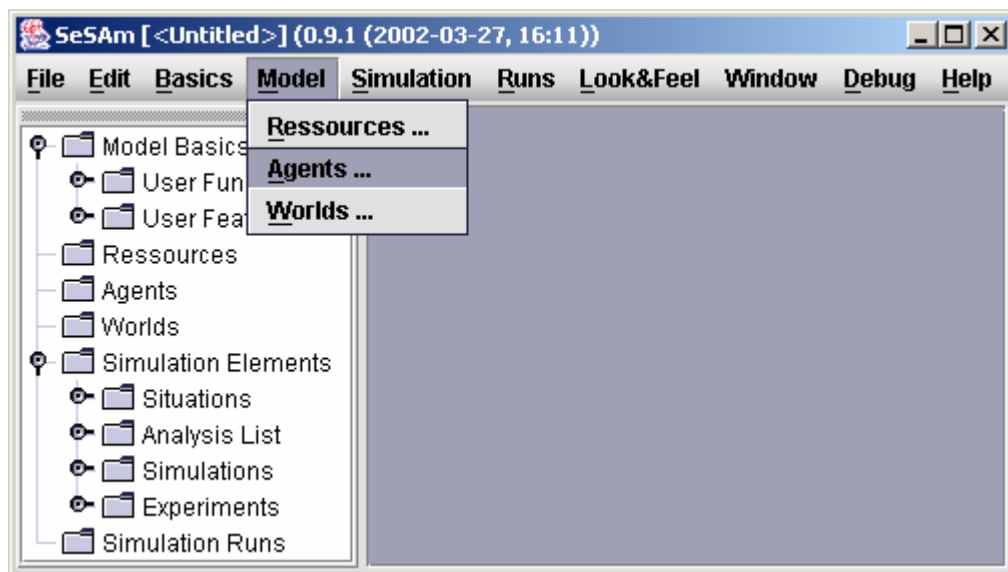


# Simple Hunter Prey Model

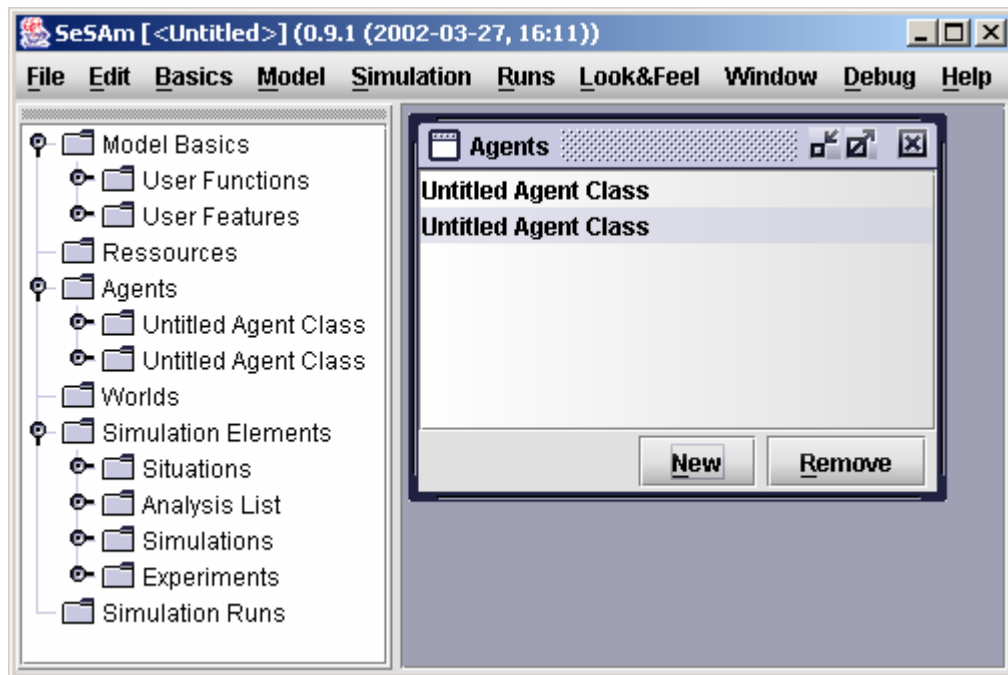
Initially, create a new model:



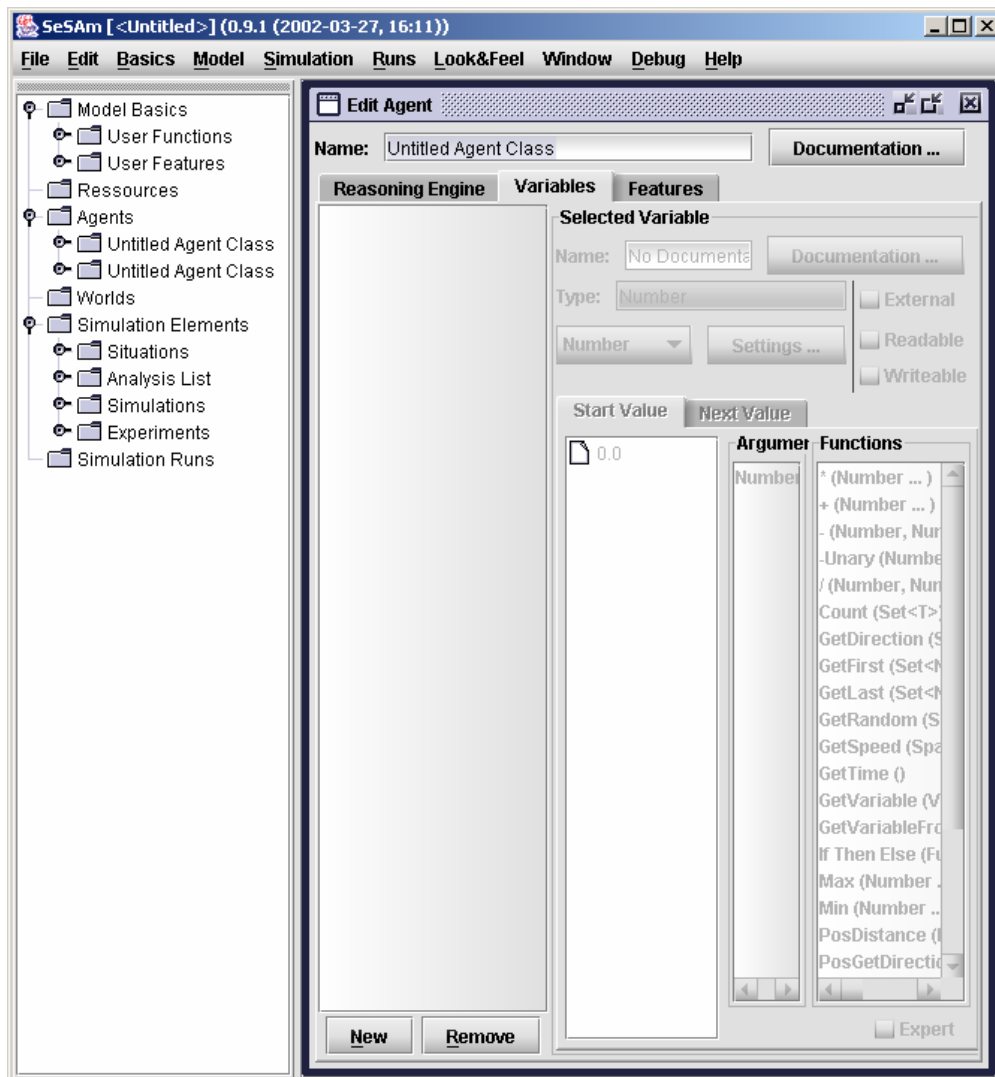
Then go to the Agents windows in the model menu ...



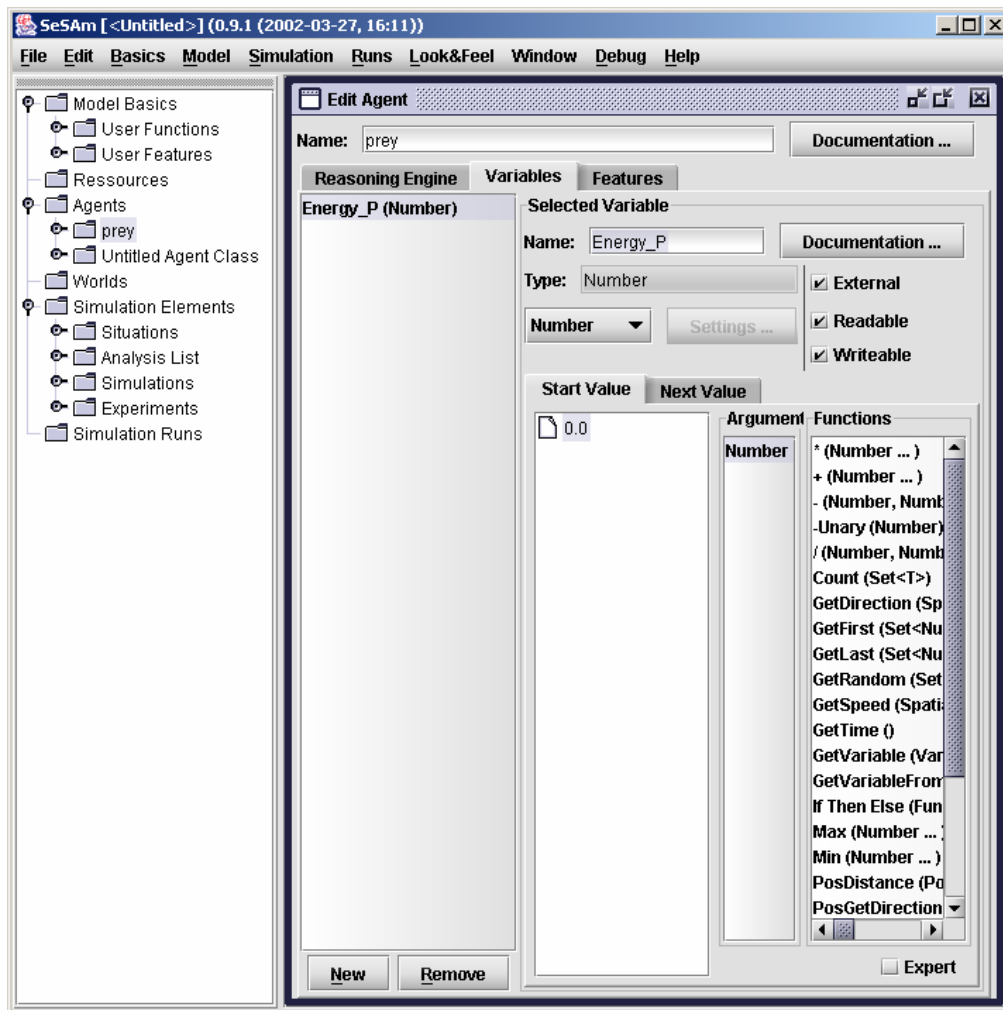
...and create two new agents



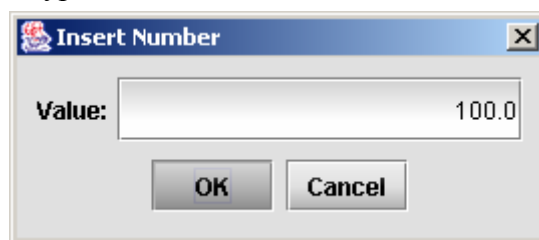
Use double click to over the option „Untitled Agent Class“ (which appears after the creation of a new agent) to open the window „Edit Agent“. In the above part of it, type the name of the agent (in this case "prey").



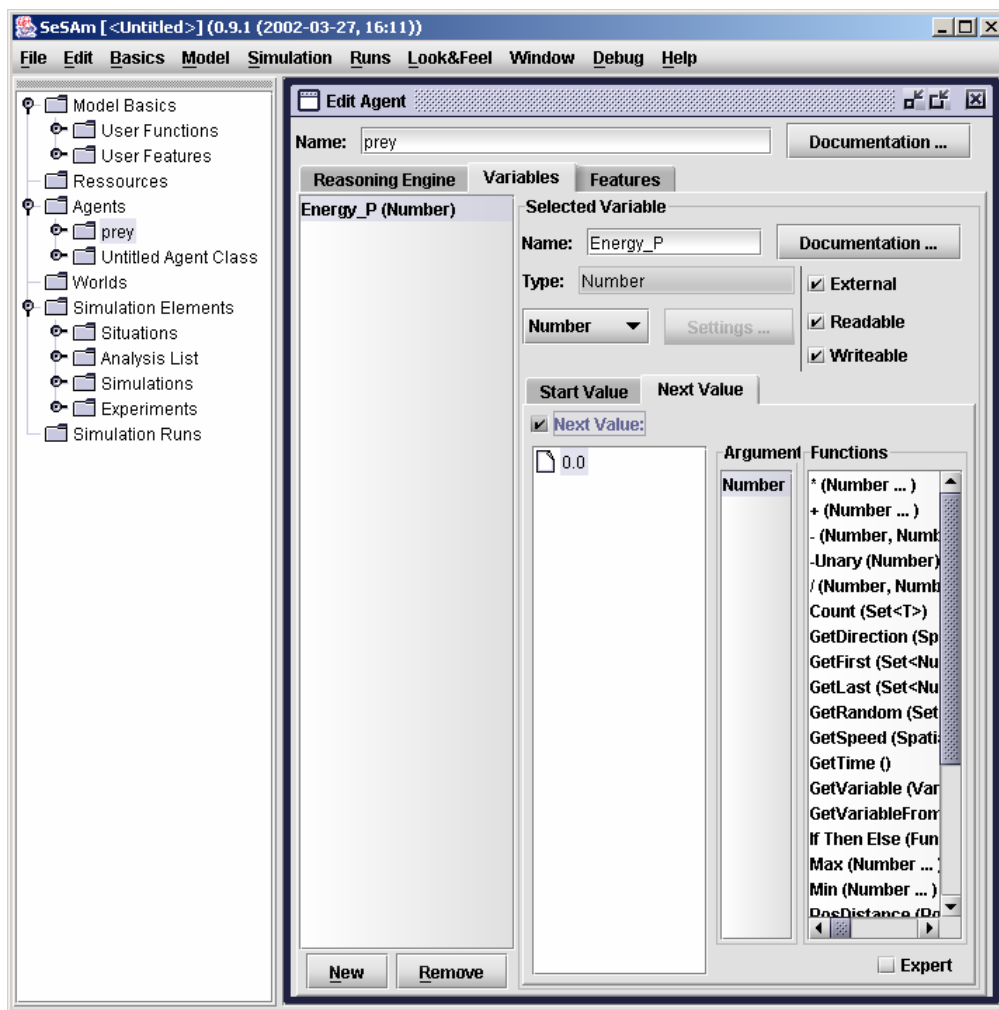
Select the option "Variables" in the menu below the line you just typed the name, and use the option "New" to create a new variable and give a name to it such as „Energy\_P” (energy of prey). It is important to click the boxes „External“, „Readable“ and „Writeable“, otherwise the variable will not be changed or updated.



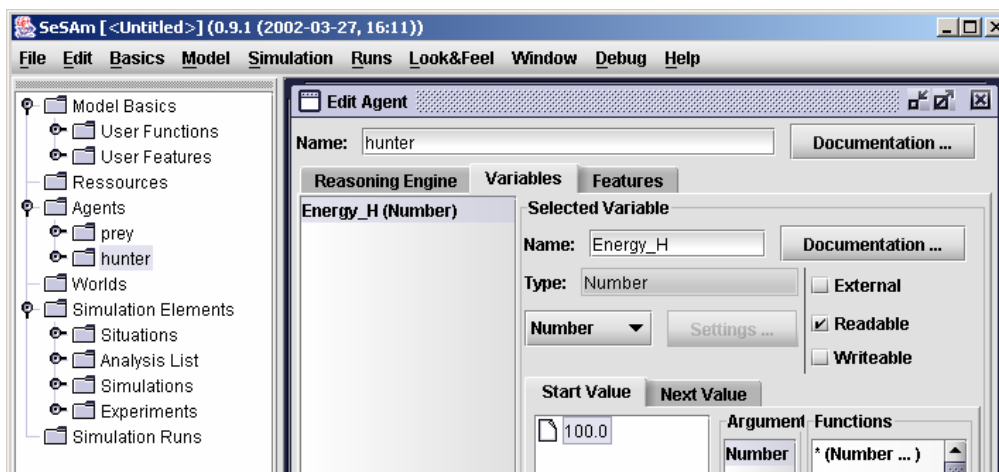
Next, you should double click over „Start Value“. A new window appears in which the start value of the variable can be typed. Please confirm with „OK“.



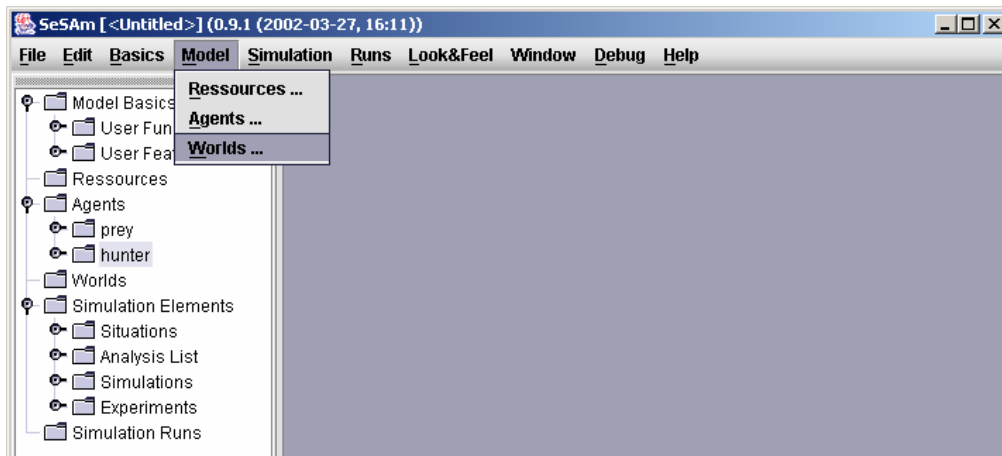
If you want the value to change automatically within time, you have to double click over „Next Value“ and either type the next value or type a function which describes the change of value within time (not necessary for this model).



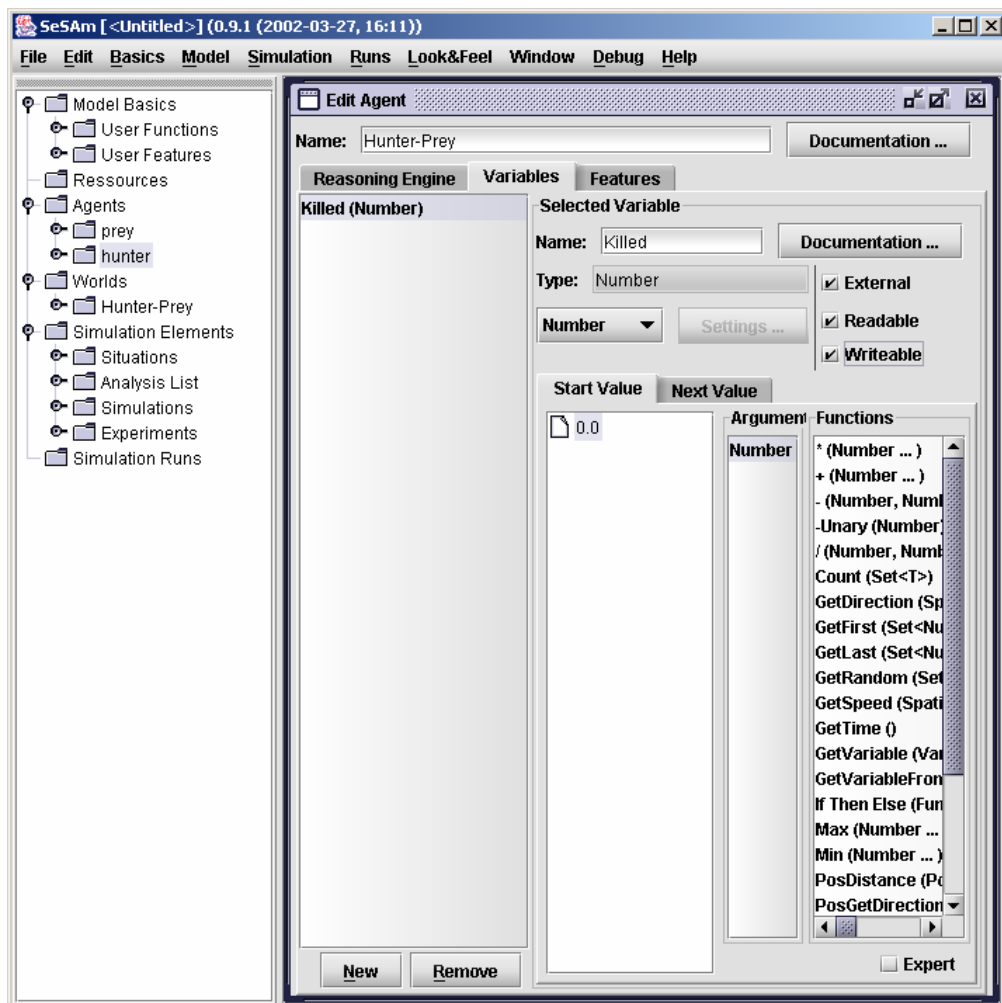
After you close the window „Edit Agent“, you should repeat the procedure for the second agent (call it „hunter“) and use the following „Start Value“ = „100.0“:



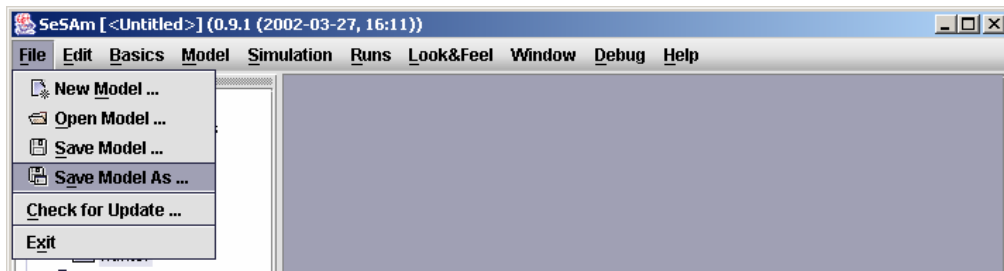
Once both agents are created, you must go to the option „Worlds“ in the „Model“-menu and create a new world:



With double click over „Untitled World Class“ you will get the window „Edit World“. Here you can change the name of the environment as for instance „Hunter-Prey“. Also, it is possible to click over the option „Variables“ to edit the variables of this world or environment. For instance, as shown in the figure, the variable „Killed“ is created, which has type „Number“ and start value „0.0“. Please do not forget to check all 3 boxes („Readable“, „Writeable“ and „External“).

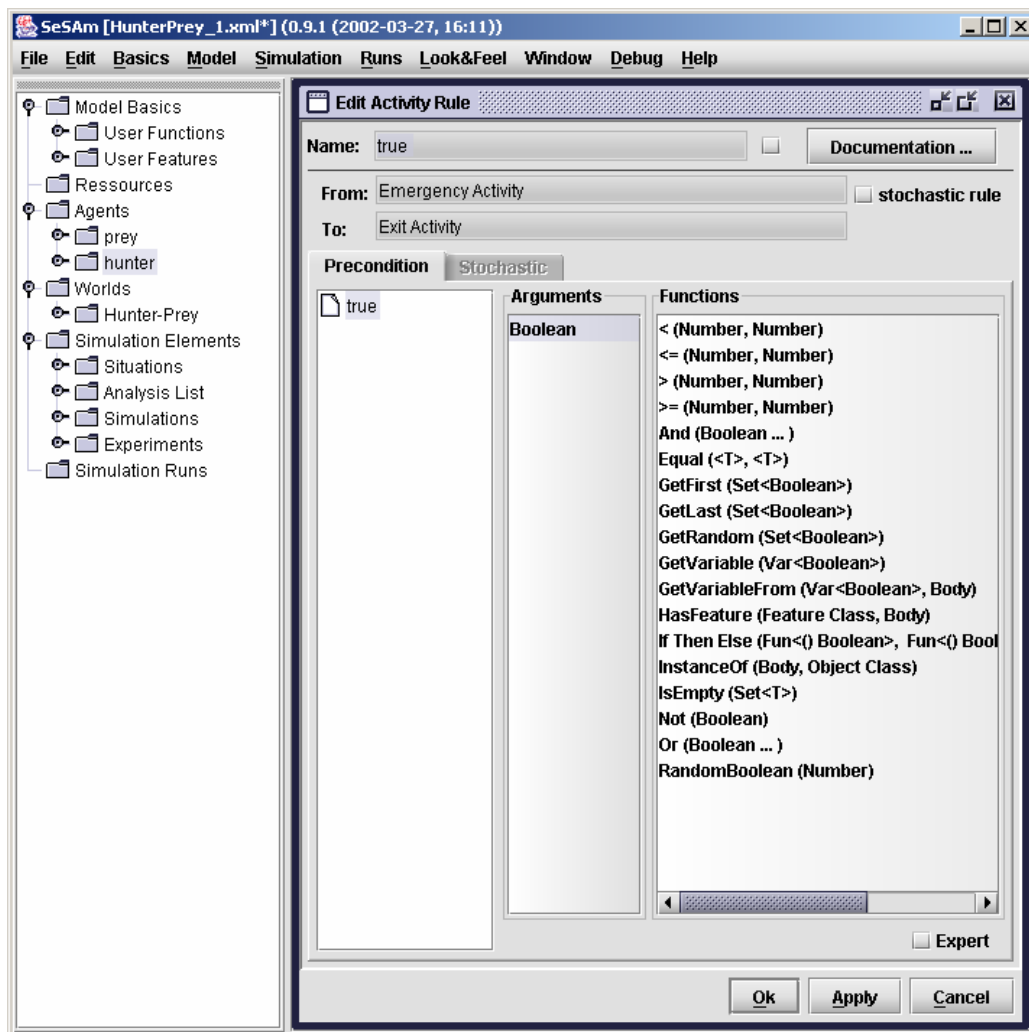


It is advisable to save the model at this point (or also after the creation of the agents). To do this, select „File“ and „Save Model As ...“. In the example, the model was saved with the name „HunterPrey\_1“:

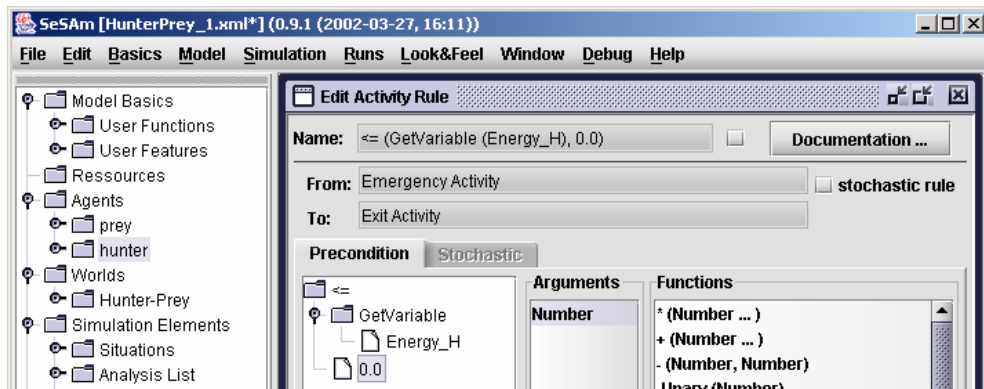


Only after the model is saved you can start working with the option „Reasoning Engine“. To do this, first double click over the desired agent (for instance, first with the class „hunter“). Agents can be accessed by the list given in the window „Edit Agent“. To set the conditions associated with the objects from class „hunter“ (no matter the state of this object), do as follows. For instance, if you want to set that if the value of variable „Energy\_H“ is less or equal than „5.0“, then the object dies, first you have to „add a rule“ by means of an "rule arrow" which links an „Emergency“-Activity with an end activity.

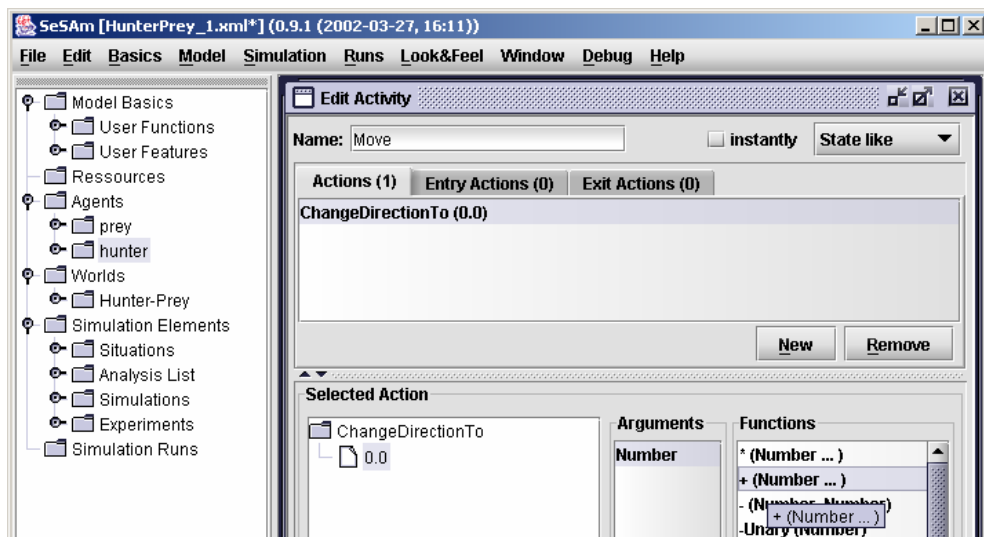
Double click over the just created rule to edit it (change).



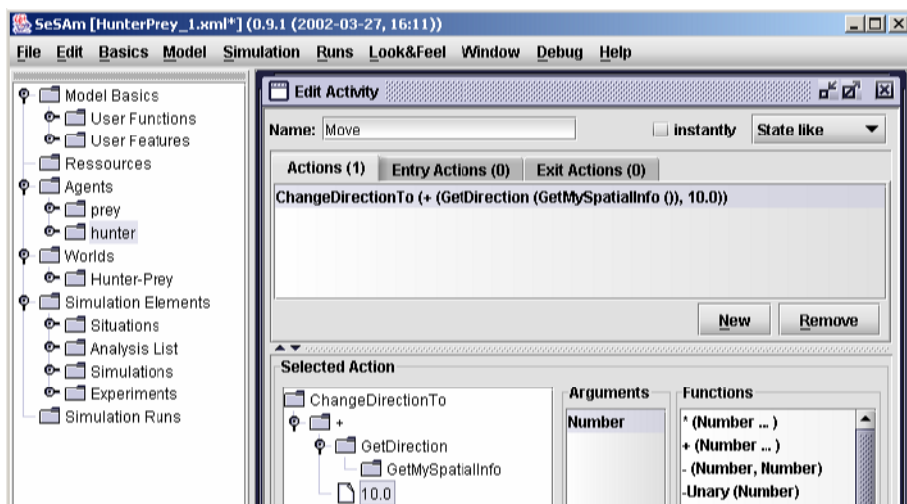
If you use the functions „<=“ and „GetVariable“, and select the variable „Energy\_H“ as first argument, then the condition „Energy\_H <= 0.0“ is created. The only thing to do now is to change the value to "5.0".



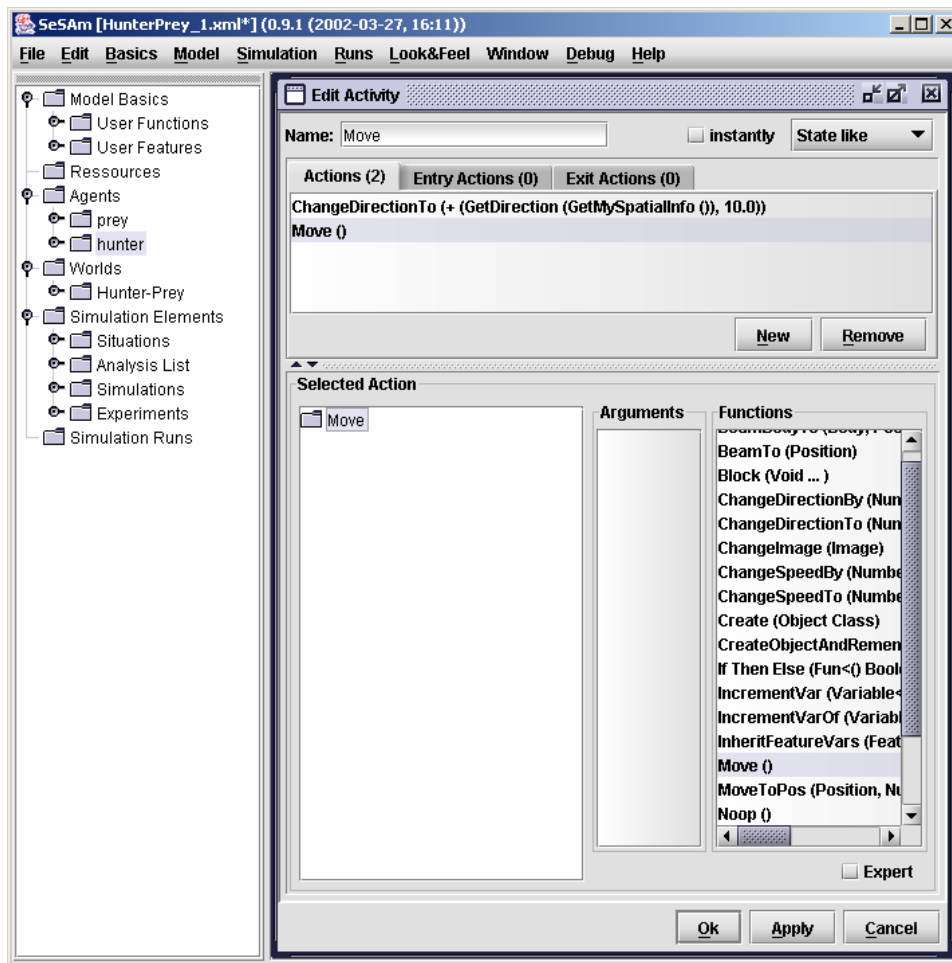
After, you should change the „Atomic Activity“ to „Idle“ by double clicking on it. It is also possible to define one-time actions by selecting „Entry Action“ and „Exit Actions“. Other actions can be defined which are performed each time during the existence of the agent. For instance, if you want an agent to change its direction by a value of „+10.0“ each time step, then you can select the following function: „ChangeDirectionTo“:



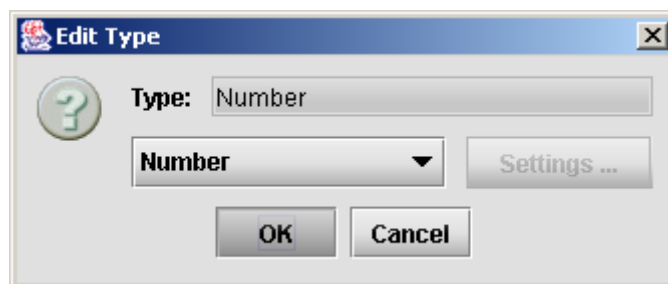
Using the add-function „+“ on a previous direction (which you can get by means of „GetDirection“ as the first argument), and the value „10.0“ (as second argument) is shown below:



With the option „New“ one can produce a new action and use it to change the „Move“ action:

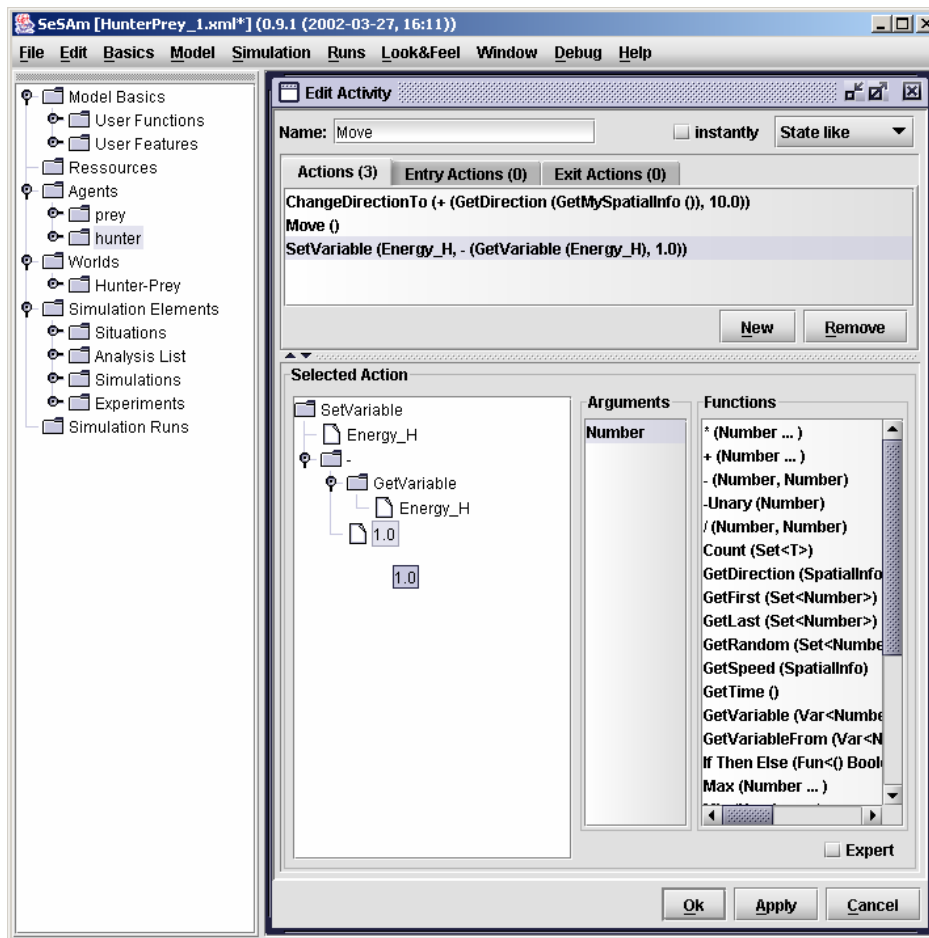


Now you probably want to define some actions and use the option „SetVariable“. If you click (double) over this option you can change the type of the variable selecting from a list (for instance to type "Number" as you see in the figure). After you should confirm with „OK“.

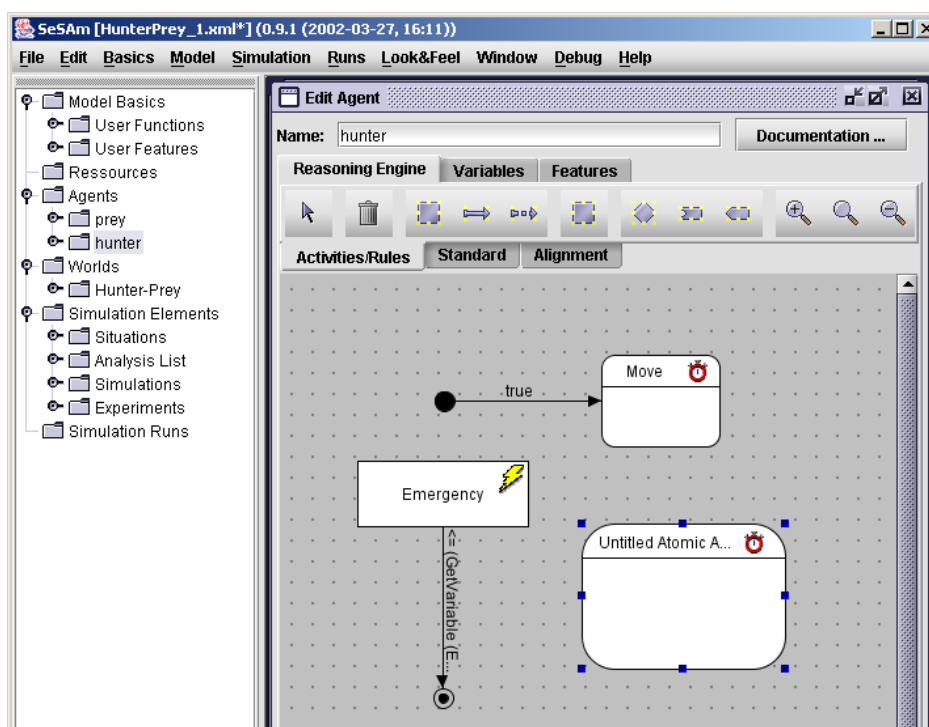




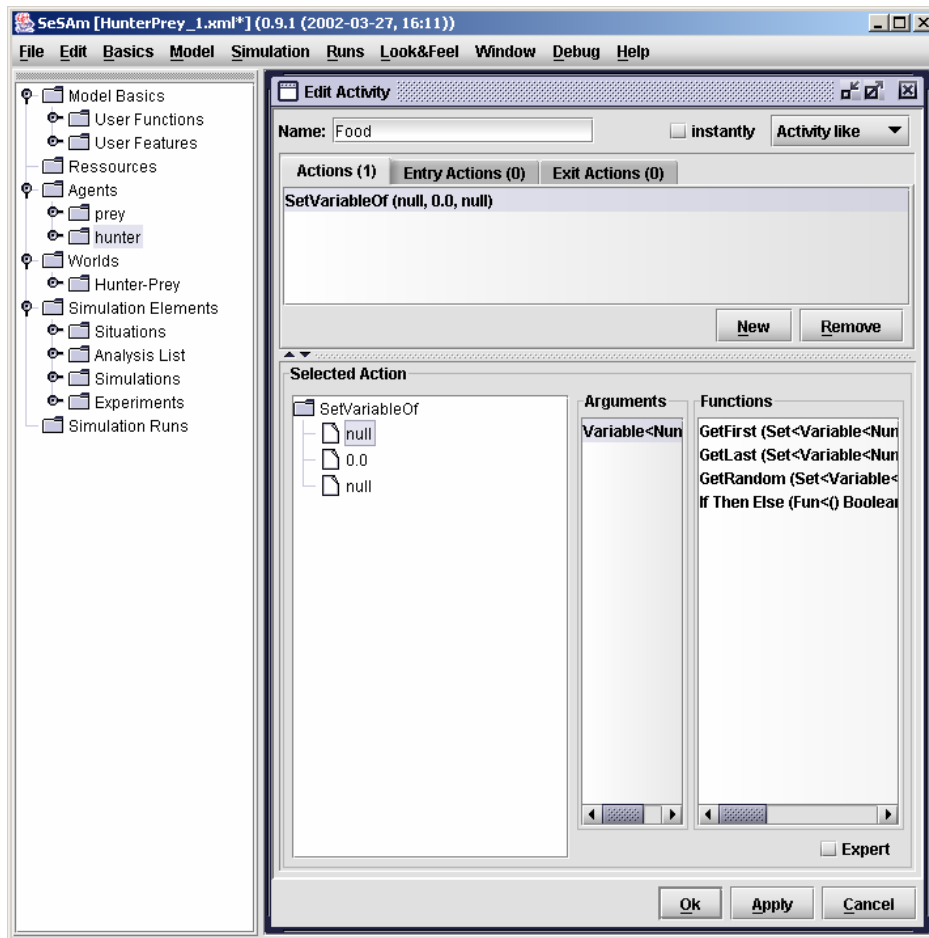
You can update the previous value of the variable by subtracting (for instance minus „1“) with („-“) and setting it as a new value:



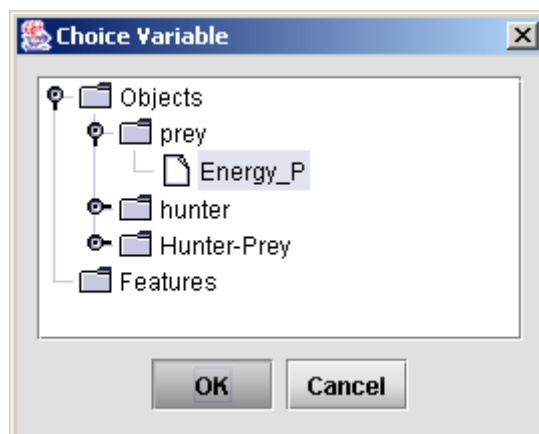
At this point the agent „hunter“ can start moving. Since it needs energy (i.e. it must get a prey agent), it is necessary to create a further „atomic activity“:



By double clicking you can change the just created activity (for instance to change the name to „Food“). Since the variable „Energy\_P“ of agent „prey“ has to be minimized (i.e. this agent must be eaten), it is necessary to use the „SetVariableOf“:



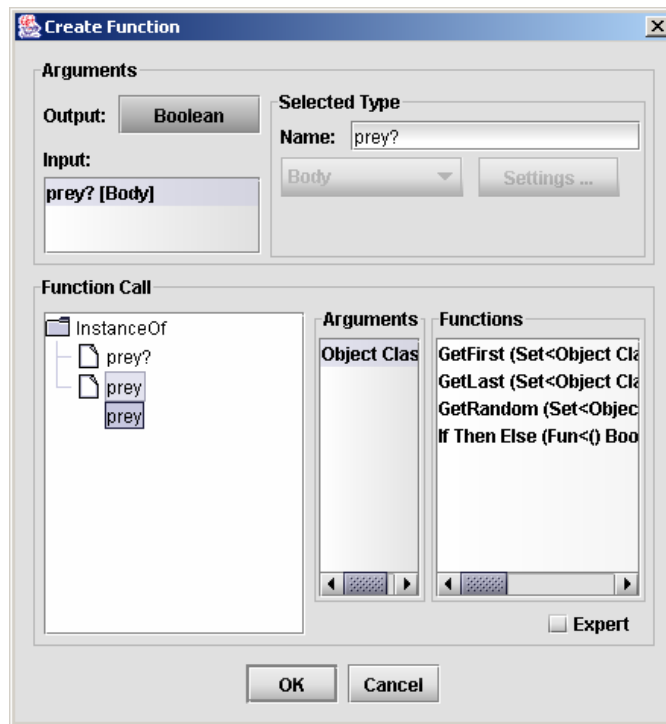
The first argument (access by double click) must be passed to the other variable:



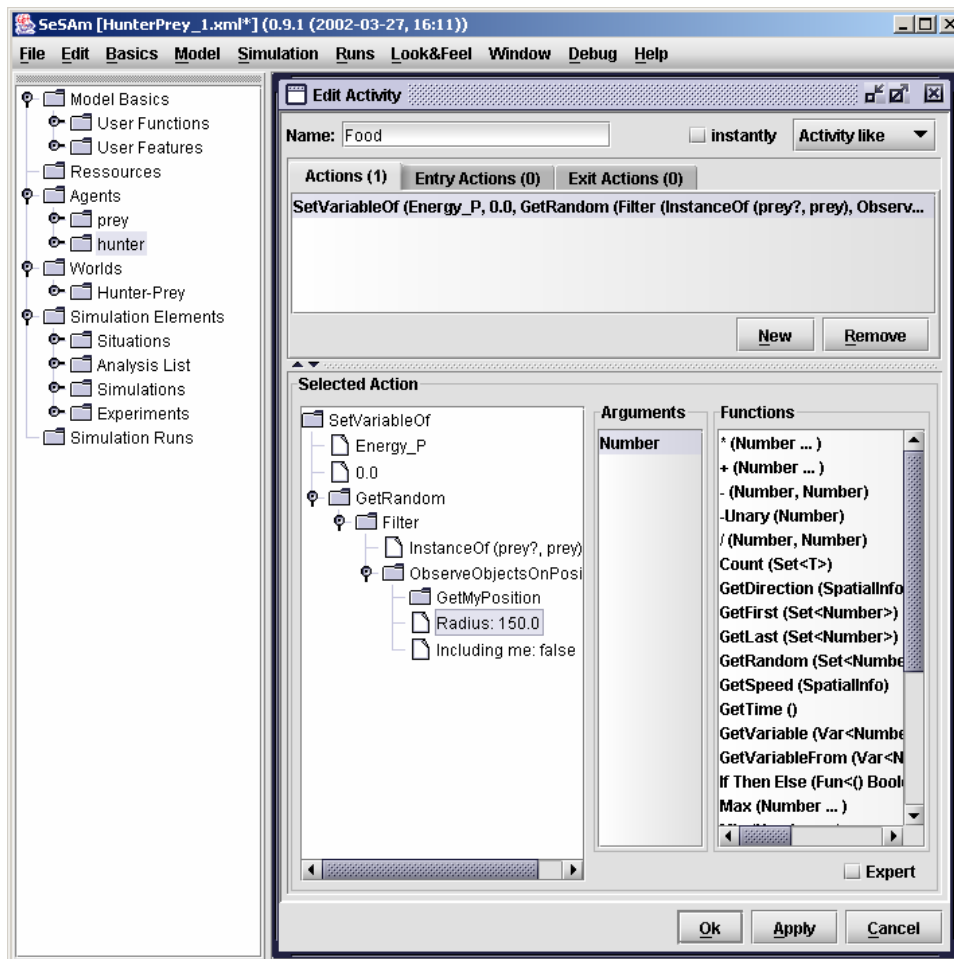
Now the object whose variable has to be changed can be displayed. Use „GetRandom“ to select a random object from the list. The conditions which appear in the list can be filtered. To change this, double click over the first argument „Function (Body) Boolean“ („Function (SimObject) Boolean“ in newer versions) of the filter.

The condition related to this example must be that only the object „prey“ is in the list. This can be achieved with the use of „InstanceOf“, in which the first argument should be set to

„Untitled (Input)“ and the second argument is an instance of type „Prey“. Then, confirm with „OK“:

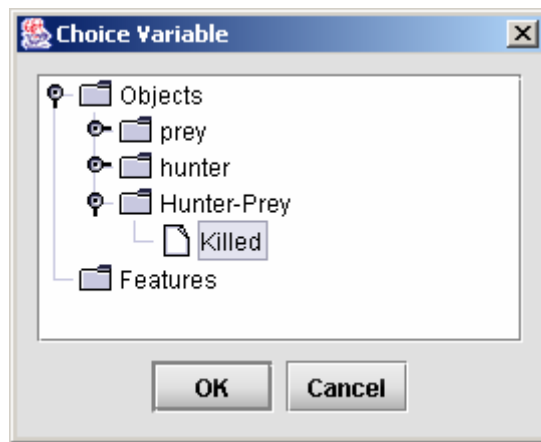


One can choose how to produce the list by means of the second argument of the filter („ObserveObjectsOnPosition“). Please change the observation horizon to „150.0“ (the objects have horizons of 100.0, which are later represented by small boxes in the map).

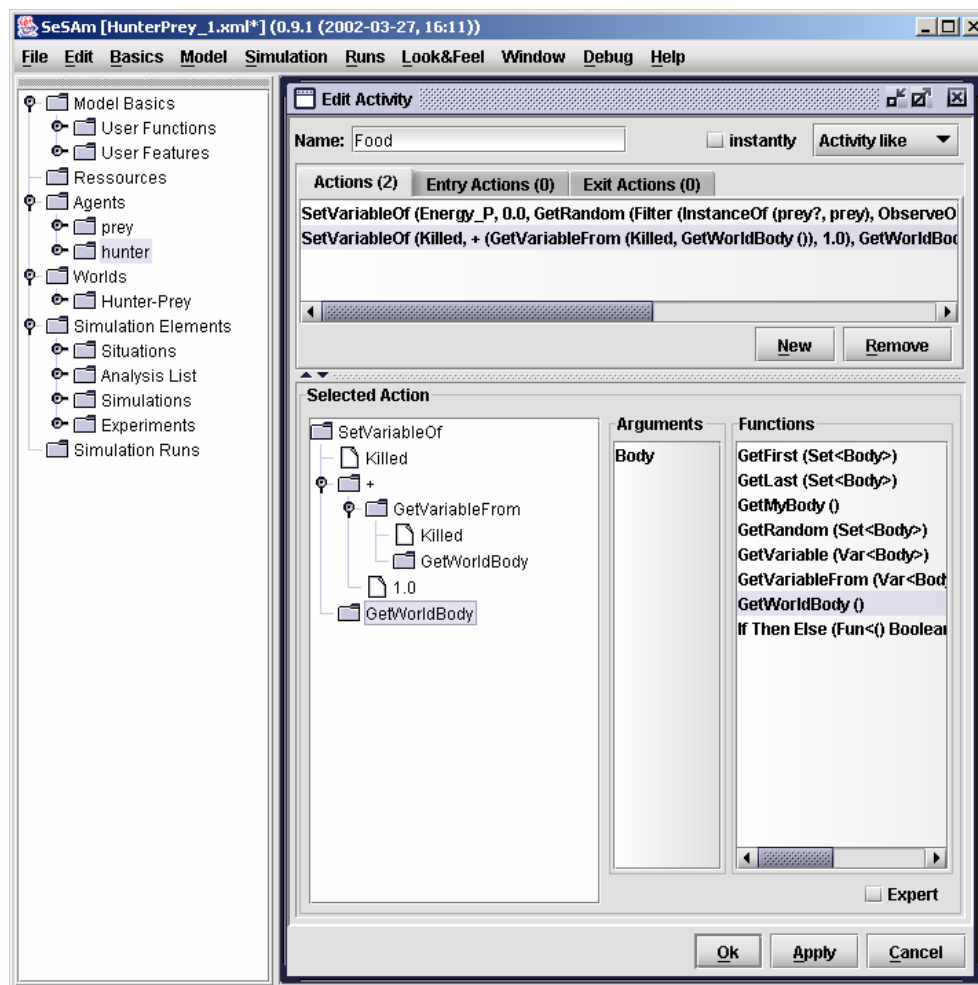




Besides, it is necessary to define a new action (with „New“) to increase the number of death preys by one (variable „Killed“ from class „Hunter-Prey“). This is done by means of the function „SetVariableOf“. One has to select the variable to be changed (first argument of the function), which in this case is „Killed“, from the list.

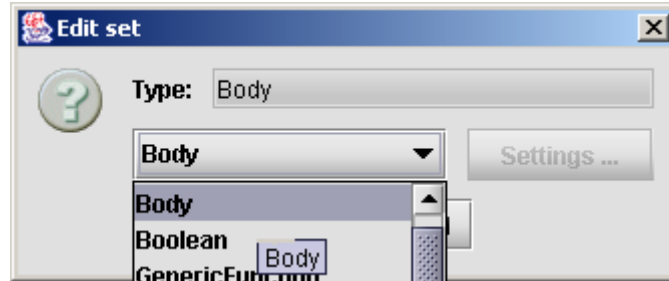


As second argument („Number: 0.0“) one can use the function „+“. Its first parameter must be a „GetVariableFrom“, since the variable „Killed“ of „GetWorldBody“(„GetWorld“ in newer versions) is to be increased. Then, the second parameter of the function „+“ is set to „1.0“ and the Body-Parameter of the function „SetVariableOf“ is similarly set by „GetWorldBody“ :

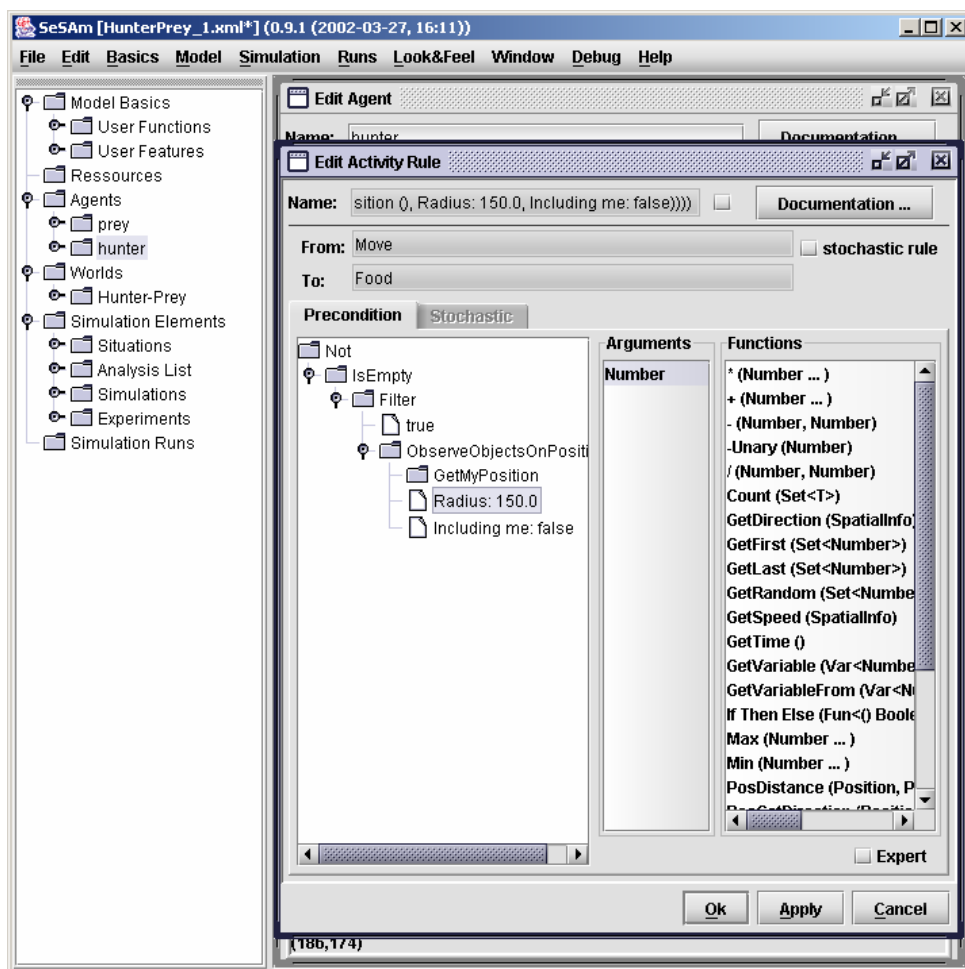


As „Exit Action“ you should then inform that, after the prey is eaten, the object has to have its variable „Energy\_H“ increased by „100.0“. This can be done with „SetVariable“ (as above).

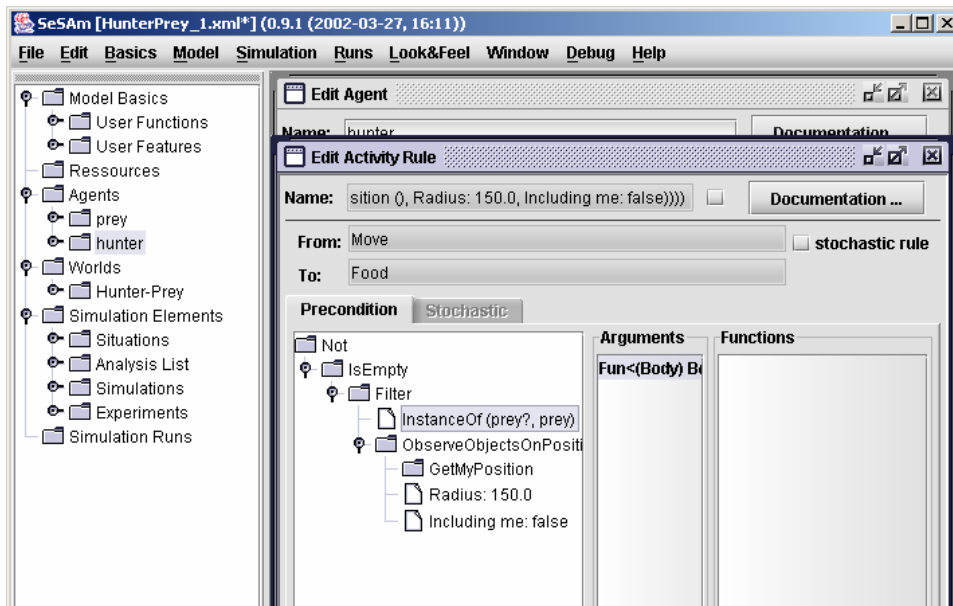
Now, the rules that define how to change from activity „Move“ to activity „Food“ have to be created. The agent „hunter“ can only ‘eat’ when its list (set Type = „Body“ or “SimObject” in newer versions) is not empty (use „Not“-„IsEmpty“),



... and there is a prey object in its horizon of „150.0“.

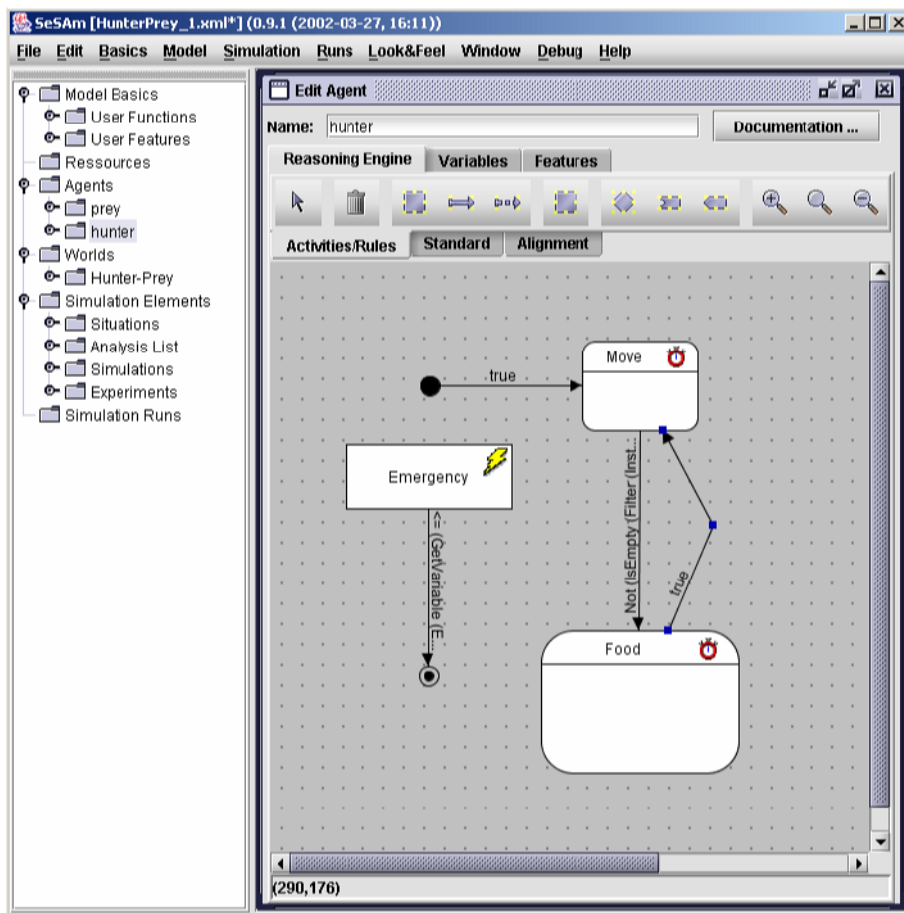


For „Set (Body)“ („Iterator (SimObject) “ in newer versions) it is possible to use the function



„Filter“ („Select“ in newer versions) as seen before.

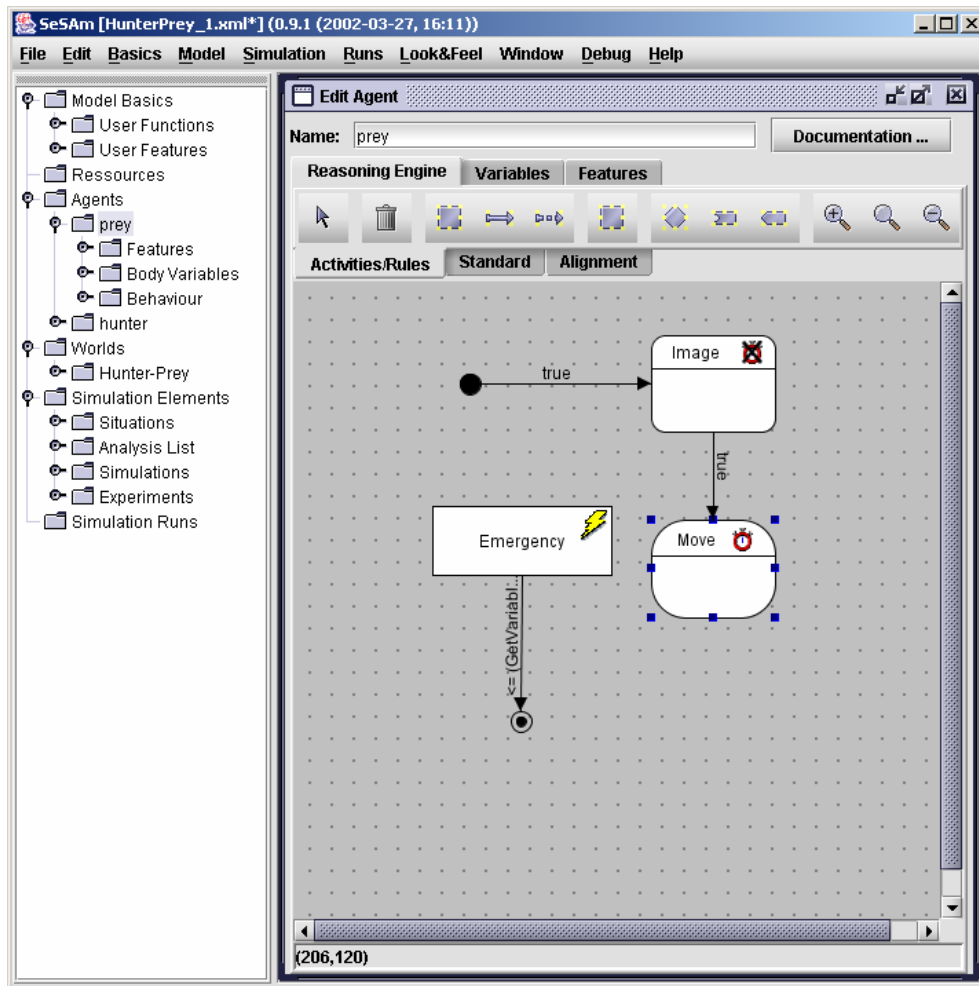
Now it is necessary to set a „true“-rule associated with "move" activity (otherwise the activity "eat" would go on forever). After that, the agent can start moving.



The agent „prey“ have to perform its end activity in case its variable „Energy\_P“ achieves a value less than „0.0“. In this case, the name of the activity changes from „Idle“ to „Image“, and the checkbox „instantly“ is activated. This activity is then performed (this does not cost

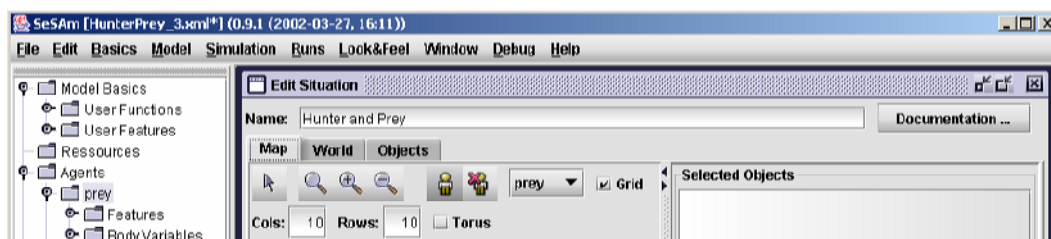
time and is performed in the same simulation step). Also associated with this activity, it is possible to change the agent's standard image (which will appear in the map). Use the function „ChangeImage“ to do this (by double clicking over „Image “ in the „Arguments“). If you click in „no image“ you can browse directories and select an image from a file and confirm with „OK“. The prey and also the hunter agents should now move. To do this, it is necessary to define new „atomic activity“:

This new activity can be called „Move“. The actions like „ChangeDirectionTo“ (for instance to increase the previous direction by „10.0“) and „Move“ are defined as above. It is still necessary to link the activity „Image“ with the activity „Move“ through the „true“-rule:



After the actions of all agents are set, you need to save the model (do this again with „File“ and „Save Model As ...“, changing the name now to „HunterPrey\_2“ or similar).

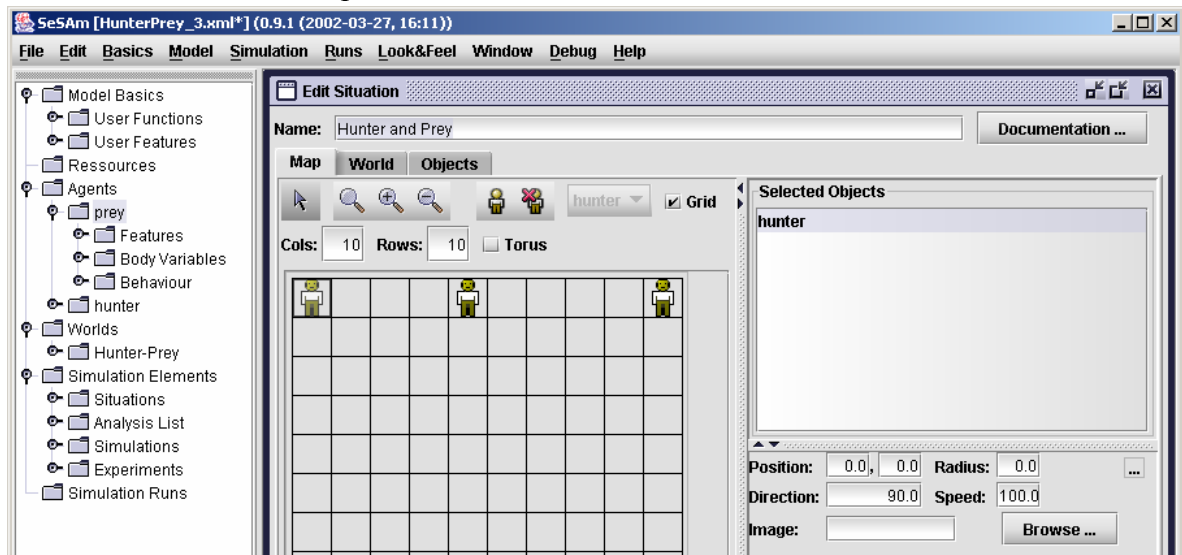
Now you can select the window „Simulations“, then the option „Situation“ and „New“ to create a new simulation. Change the name of this new simulation with double click (for instance to „Hunter and Prey“ as in the figure). Change the map with the button „Map“.



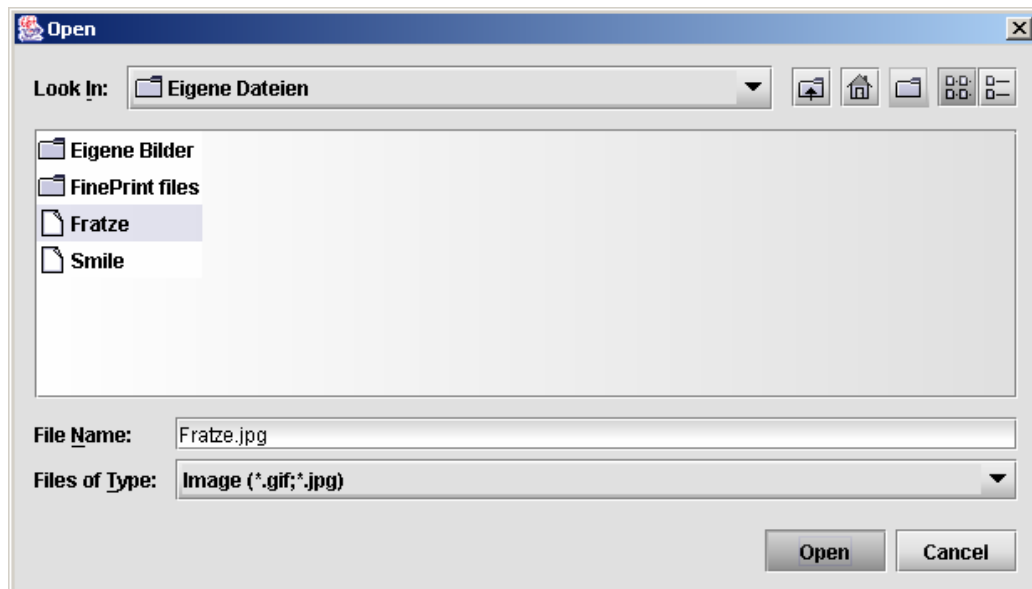


You can add an object to the map by using the button presenting a figure - respectively to remove an object of the map by using the button presenting a crossed figure - above the „torus“-checkbox. To select an object of the map use the button presenting an arrow. With „Cols“ and „Rows“ you can change the size of the map. And with the magnifier-buttons you can zoom the map. The checkbox „Grid“ allow the grid to be shown.

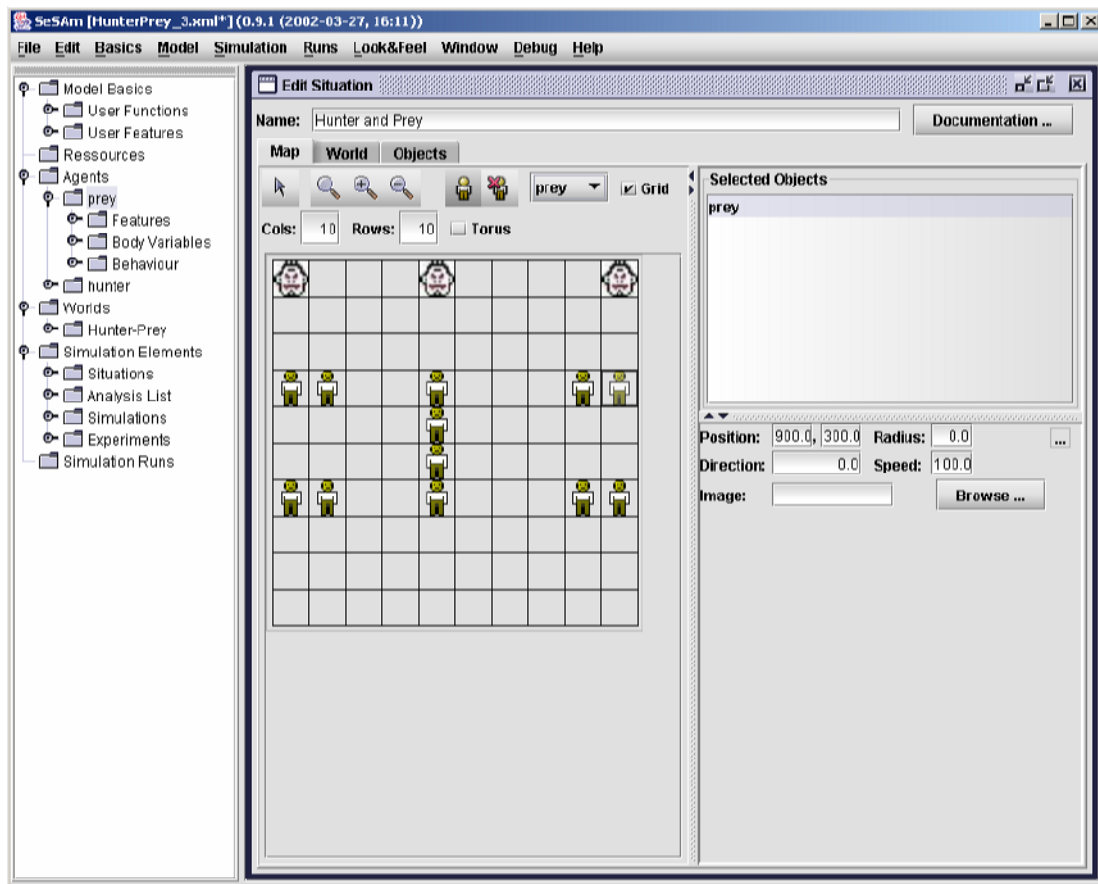
Then, please use the button presenting a figure to insert 3 „hunter“-agents and set their directions (use „Direction“ to do this) to „90.0“. To do this, first choose the agent. If the field „Direction“ is not shown, simple increase the size of the window.



Now all hunter objects get one image which can be select with the button „Browse...“:

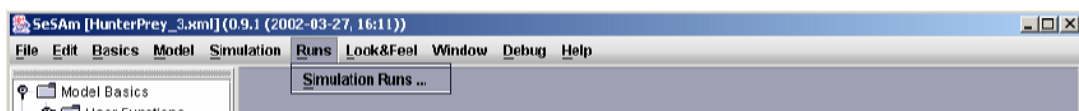


Now, insert 12 „prey“-agents. You don't need to change anything for them.

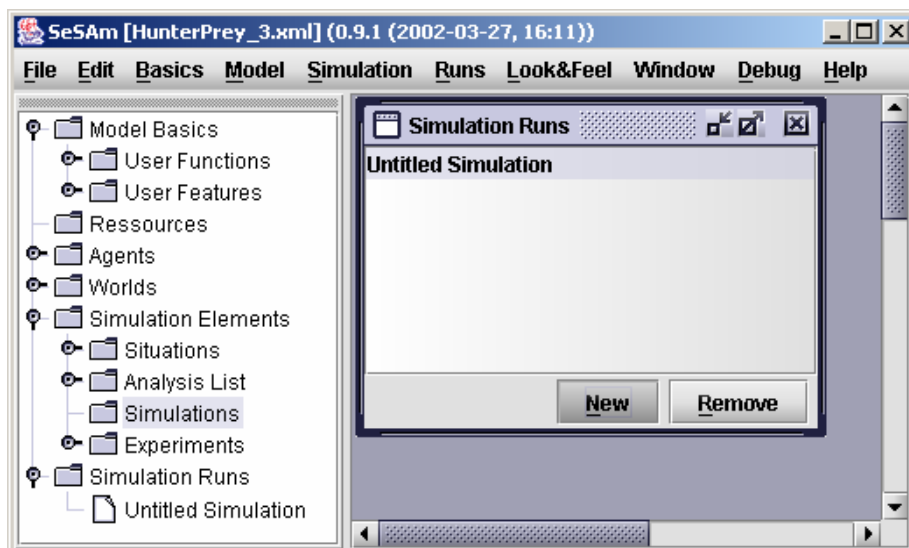


The windows „Edit Situation“ can now be closed and the model should be saved again using „Save Model As ...“ under the name of „HunterPrey\_3“ or similar.

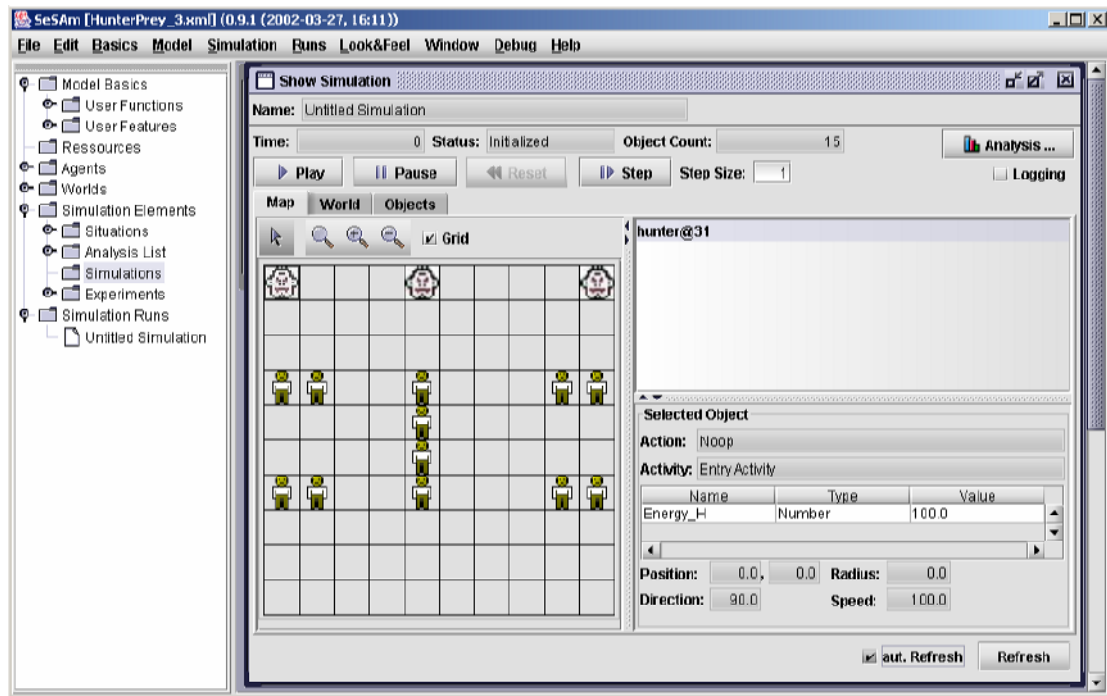
Change to the simulation runs list with „Runs“ and „Simulation Runs“.



Use „New“ to create a new simulation run.



With double click over this simulation run you can edit it (by means of the „Edit Simulation“ window). Selecting „Objects“ you can see all objects which are not in the end state. Selecting „Map“ you see all objects which are in the map. Selecting „Grid“ the grid is shown. Important: if you want to show the experiment in the updated state automatically, please activate the „aut. Refresh“-checkbox at the inferior right side:



With the buttons „Play“ and „Pause“ you can start and pause the run, with „Reset“ you can restart the experiment and with „Step“ you can run the experiment step by step

END