Processes in OpenSimpplle v1.0

The following is an explanation of the role of processes in OpenSimpplle. Processes can be a variety of things ranging from prairie dogs to blister rust and fire events to droughts. In general though, there are two basic roles that processes can play.

First is in the pathways where the users can identify what change a process makes (if any) to the "next-state" for the vegetation community. These can be input in the various dialogs of the GUI (in source much of the code for these are in the editor classes of simpplle.gui). Also, as a parallel, one can consider aquatic processes and aquatic states and what would be the "next-aquatic-state" as a result of a process occurring.

The second role is to influence all the other system knowledge that has an interface screen where "Columns" for processes can be selected from the pull down choices. There are some types of system knowledge where processes don't have an impact and these have different interface screens where "columns" is not a choice.

The intent is to let users identify the specifics of the process roles and impacts. For each zone there is a default system knowledge that came from many workshops made when the zone was first created. However these default choices are often changed for specific analysis by different sets of users and they load their own system knowledge files on top of the default logic. This overrides the default logic (in source there are boolean checks for whether the system knowledge used is default).

What role exactly processes play in any given simulation is in large part left to the user. This is entirely by design and is most often not the same as the default system knowledge. For example one set of users could say the process of mountain pine beetle in the pathways kills all of one tree species. Another set of users, in the same zone, could decide the process doesn't kill all the species, just changes size class; so the "next-state" in the pathway is different. They do both use the pathways to identify the next-state, just in fundamentally different ways. Likewise some users will want to use mixed-severity-fire process in the probability logic screen for DF-Beetle to give a different probability if MSF occurs. Then there will be some users who do not use it in the DF-Beetle logic screen. The way in which users choose to use the process varies, but they all have the ability to choose to use the process to influence other processes probability of occurring.

Note to developer's: all the relationships to processes should be mutable. If there are some instances in the code where it is hardwired, these resulted from initial development in a zone and should have been replaced with the incorporation of the choice of processes through Columns in the GUI. If a hardwired case still exists it is something that was overlooked / forgot and should be replaced in a future release. **As of OpenSimpplle v1.0 none are known. Any that come to attention follow the contact information below and let the administrator know.