



AnyLogic 6.4.1

New Features

© 2009 XJ Technologies Company Ltd. All Rights Reserved
AnyLogic and XJ Technologies are registered trademarks of XJ Technologies Company



Contents

- New type of license: University Researcher 3
- More functionality included in the Educational License 3
- Import of Vensim™ models 3
- Other improvements for System Dynamics modelers 5
- Performance improvements in Network Based modeling 5
- Other improvements in the Enterprise Library 5
- Other Engine improvements 6

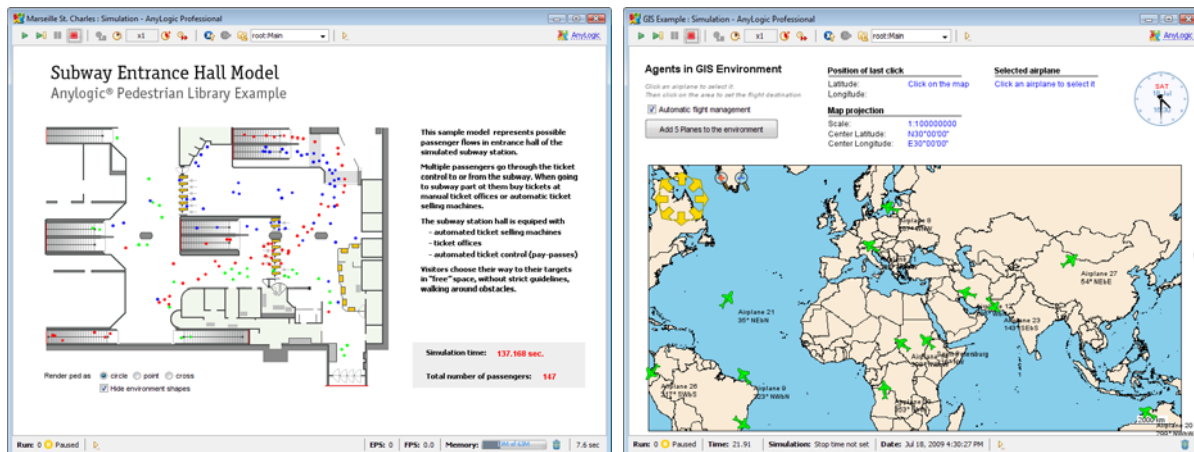
New type of license: University Researcher

To give university-based researchers maximum functionality at an affordable price, we are introducing a new type of license – the University Researcher license, which can be considered as analogue of AnyLogic Professional for researchers. This license can be used by those who perform public research, and includes more features than AnyLogic Advanced, in particular:

- A limited version of the Pedestrian Library
- Fully functional OptQuest optimizer
- Integration of GIS maps into Anylogic models
- Full-featured Java-level and event-level debugger
- Extended set of experiments (Compare runs, Monte Carlo, Sensitivity analysis, etc.)

More functionality included in the Educational License

AnyLogic 6.4.1 Educational license now also includes a limited version of Pedestrian Library and integration of GIS maps. The Educational EULA is limited to teaching.

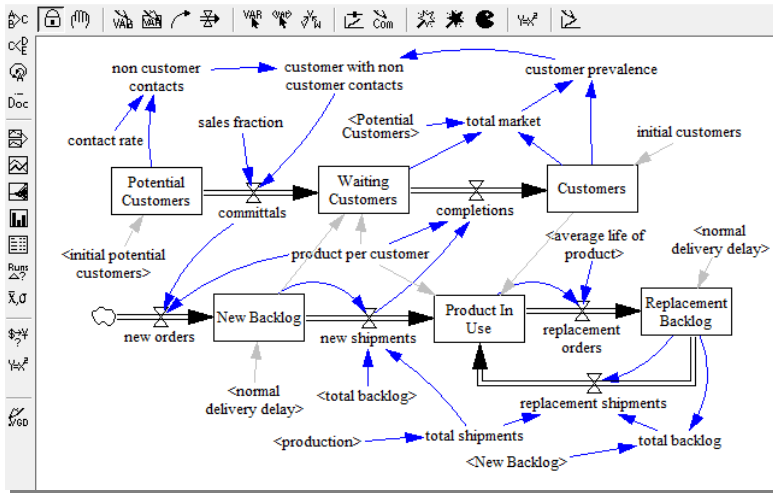


**Pedestrian dynamics modeling and GIS integration is now available
in Educational and University Researcher licenses**

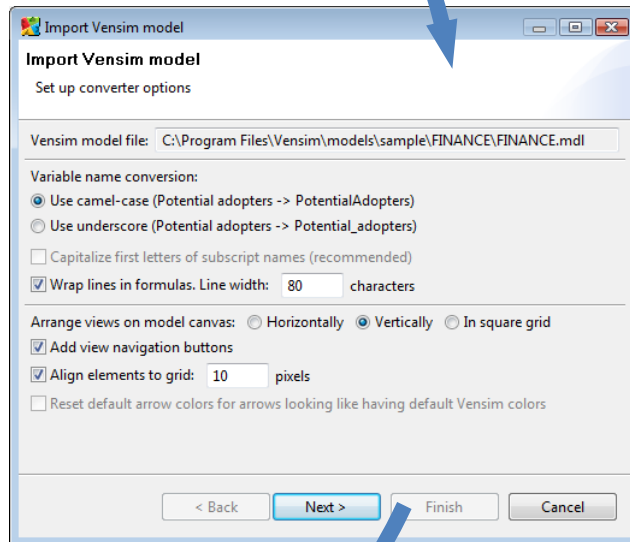
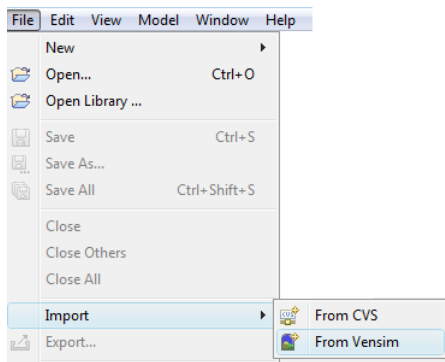
Import of Vensim™ models

Starting with version 6.4.1 Anylogic supports the importing of Vensim™ models. A new model converter reads a model in .mdl format and creates a ready-to-run equivalent AnyLogic model. The converter supports virtually all constructs of Vensim™ modeling language including subscripts, special functions, etc. Vensim™ views become AnyLogic View Areas.

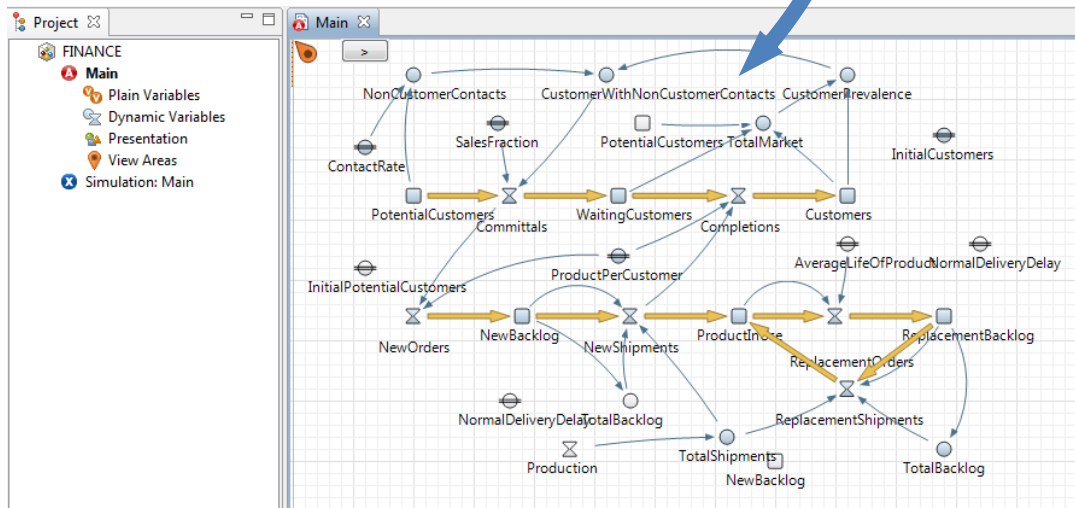
The converter allows users to transfer existing Vensim™ to AnyLogic to combine them with agent based or discrete event models, to add sophisticated animation, to publish them as Java applets on the web, or to integrate with other software as Java applications.



Vensim™ model



AnyLogic model



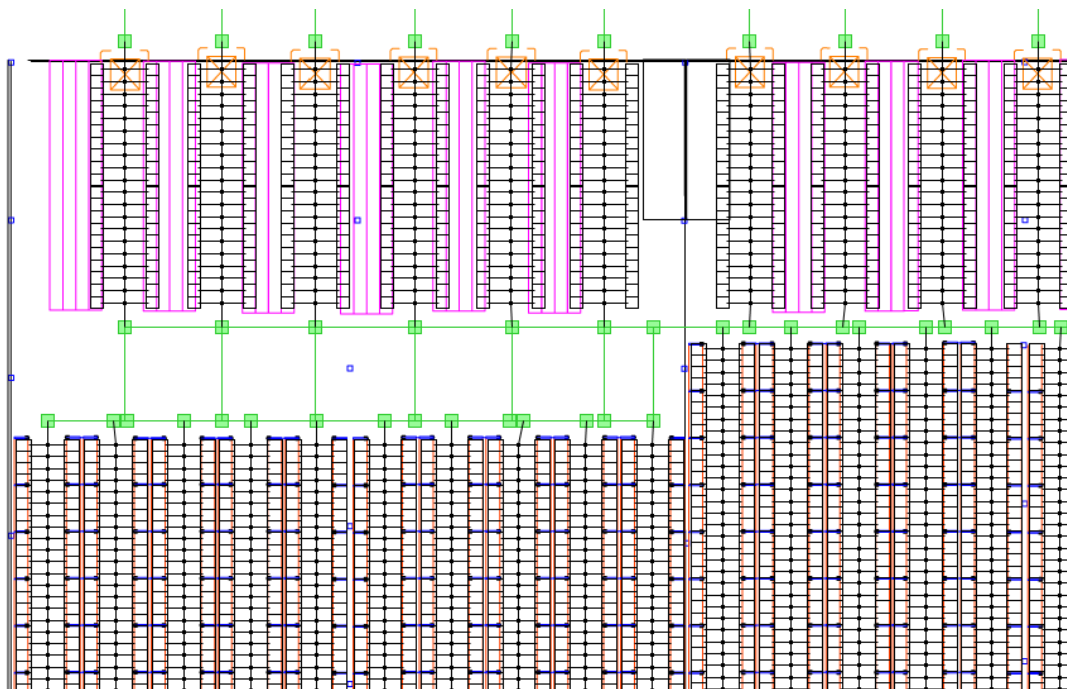
Importing Vensim™ models

Other improvements for System Dynamics modelers

- delay() function now supports variable delay time.
- A set of new functions delayInformation(), delayMaterial(), delay1(), delay3(), smooth(), etc. are now available in the formulas of dynamic variables.
- Dynamic variables marked as constants are now initialized correctly even if there are algebraic loops in their initial values. They may also be changed during discrete events.
- Performance of both runtime and code generation of system dynamics models has been significantly improved.

Performance improvements in Network Based modeling

Networks with a large number of nodes (thousands) are now initialized faster and consume less memory than before. To achieve the maximum possible execution speed of network based models, AnyLogic prepares routing and distance information in advance during the network initialization phase. The current optimization was focused on reducing the time needed to prepare that information and the memory required to store it. This is particularly important for models where parts of the network are generated automatically, e.g. warehouse models with NetworkStorage objects.



A part of network automatically generated by AnyLogic in a warehouse model

Other improvements in the Enterprise Library

- The Network object now has a method transferResourceUnit(Entity from, Entity to, NetworkResourcePool pool). This method is provided for more flexible management of network resources. In particular, you can explicitly use one resource unit for a number of operations with multiple different entities. Or you can send a resource unit that is no

longer needed by the entity to a particular location – and the entity does not have to wait for completion.

- All objects that provide access to regular resources (Seize, Release, Service, and Assembler) now accept “0” as resource quantity. The requests are immediately satisfied in this case and the entity passes through. This can make your flowcharts more compact: you do not have to create special branches for entities that do not need resources.
- You can now fine tune the position at which entities are shown when stored in NetworkStorage objects: the parameter drawMode has been added.
- A new object Assembler implements assembly of a new entity from a number of other entities from up to five different sources according to a bill of materials. Time delay and resource utilization can be associated with the assembly process.
- The Utilities project (which can be found among the example models) now includes two new objects TimeMeasureStart and TimeMeasureEnd, which collect statistics on the time the entities spend between two points in the flowchart. An example for the usage of these objects is provided.

Other Engine improvements

- You can now use the words “infinity” (or “+infinity”) and “-infinity” in numeric expressions (instead of writing Double.POSITIVE_INFINITY or Double.NEGATIVE_INFINITY)
- The method receive(msg) has been added to the Agent. This method performs immediate delivery of a message to the agent from anywhere.