



Thank you for using Simio. We expect that you will be pleased with its power, flexibility, and ease of use.

We are very pleased to bring you **Simio Release 6**. Simio continues to aggressively introduce innovations that are changing the landscape of simulation and stimulating new development across the industry. In particular you might look at our newly enhanced Flow Library, Crane Library, and at our Risk-based Planning and Scheduling (RPS).

Many of our customers have cited Simio's flexibility, ease-of-use, and great support among the reasons why they have adopted Simio -- often switching so they can do things that are difficult or impossible with other products. But we are not resting on that success -- we are continuing to make Simio easier to use and more flexible, while adding still more technology breakthroughs.

This document describes the wealth of new features that have been recently added. Simio is ready to take on your most demanding modeling tasks.

The desire to teach using the latest innovations, combined with Simio's comprehensive academic package are moving Simio towards becoming the most widely taught simulation package with over 600 academic institutions worldwide already adopting Simio.

We are anxious to hear about your experiences. Please post your feedback in the forums mentioned below or send it directly to me. Thanks again for choosing Simio.

Dave Sturrock
Vice President – Operations
dsturrock@simio.com

Simio Release Notes

Support


The best way to get support is to use the **Simio User's Forum** found at www.simio.com/forums, or contact us at support@simio.com using one of the links on the Support Ribbon. Sign up on the user's forum as a **Simio Insider** to get full access to find product information, a place to post problems and questions, and the opportunity to engage in discussions with other users and the Simio team.

You can also find Simio User's Groups on:

LinkedIn (www.linkedin.com/groupinvitation?groupId=55167) and

Facebook (www.facebook.com/groups/13863832711).

Please visit our simulation community resources web page (www.simio.com/resources.html) for additional information.

The **Support Ribbon** (the Support tab) of the Simio software contains shortcuts to the above, plus much more. Don't miss the links to our on-line documentation, training, videos, downloads, and other helpful resources. While Simio includes comprehensive on-line help available at the touch of "F1" or  in the product, the numerous books and free video training courses provide a great way to get started using Simio.

Unfortunately we cannot provide support to students. We encourage students to work through their instructors, or to reach out to the broader user community using the Simio Insiders User Forum.

What is a Sprint?

We are following an agile methodology in which we develop in three week cycles called sprints. At the end of each sprint we have tested, documented, distributable software including new functionality. The following pages describe the enhancements added during each sprint. It is not **necessary** that you download and install every Sprint unless you are interested in a particular feature. But we encourage you to do so when convenient so that you are always working with the best software, documentation and examples we can deliver. Most sprints are posted to the Simio Insiders download area. The public releases (discussed below) are updated approximately quarterly.

Where to Find New Releases

The latest public software releases can be found at: www.simio.com/download. Your activation will continue to work with new releases as long as your maintenance is current. Newer releases are often available on the [Simio Insiders forum](#).

Simio Release 6 – Sprint 97 – November 15, 2013

In this release, we have enhanced our RPSEExample model and added two new SimBits demonstrating our new Dashboard Reports features. We've also made some stabilization enhancements, as noted below, and bug fixes. We are excited about all the new features that have made it in to Simio 6, as noted in the last several sprints' release notes sections – we hope you are too!

Dashboard Reports SimBits

We have added two new SimBits that display the new Dashboard Reports features available to Simio Enterprise and Team users.

- **DashboardReportTallies.spfx** –This SimBit takes the original TallyStatisticsInTables project and includes a dashboard report plot for one of the entity type tallies in the system.
- **DashboardReportInteractiveResults.spfx** –This SimBit is a revision of the VehicleVisitsServiceCenter project that graphically displays the resource utilization as well as a vehicle's pickup and drop off times in chart format.

Updated RPS Example

We have updated the RPSEExample.spfx project to include:

- Forklift added to transport material between material arrivals, receiving and stock.
- Server symbols updated to be more realistic.
- Worker changed to be an animated person.
- Dashboard reports added including Dispatch List and Order Details.

We have also relaxed the limits in the academic enterprise versions so the RPSEExample model will now load and run without deleting any targets.

Table Performance Enhanced

Opening a grid view (or expanding a detail view) that has a foreign key column where there were a lot of foreign key selections should now be much faster. Combo drop downs in general should be faster as well and you should no longer see 'flashing' as you open a combo with a large number of items.

Properties of Top Level Model

The properties of a top level model are streamlined -- many that were rarely used are now hidden. You can still get to them by modifying their visibility in the model's properties window.

Memory Usage for States

We now use much less memory for non-scalar states (e.g., vectors, matrix).

Simio Release 6 (Beta) – Sprint 96 – November 1, 2013

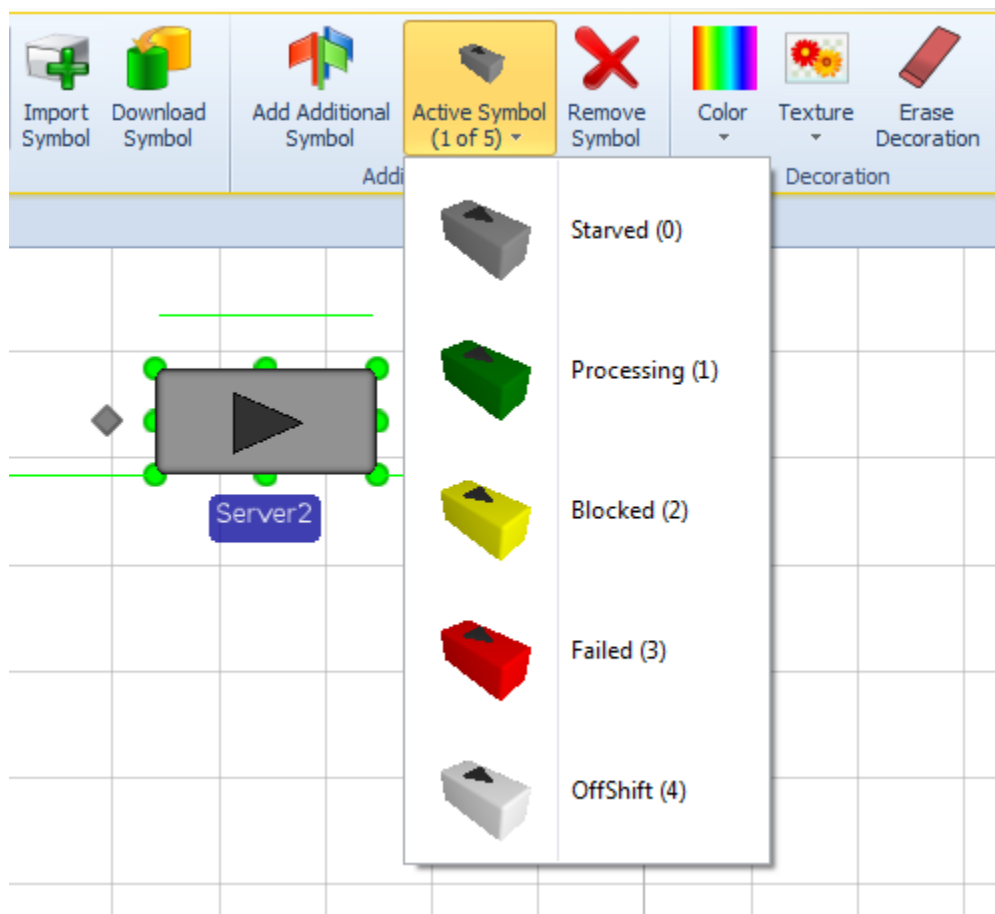
In this release, we added many valuable features. We have enhanced our Flow Library to include an Emptier and added a new Dashboard feature for greater output analysis. We've added a number of new features based on customer requests, including table property functions, multiple symbols for Standard Library objects, and improved entity-viewing features to name a few.

This software is the final Beta release of Simio 6 software and we appreciate any early feedback you may have.

Multiple Symbols on Standard Library and Flow Library Objects

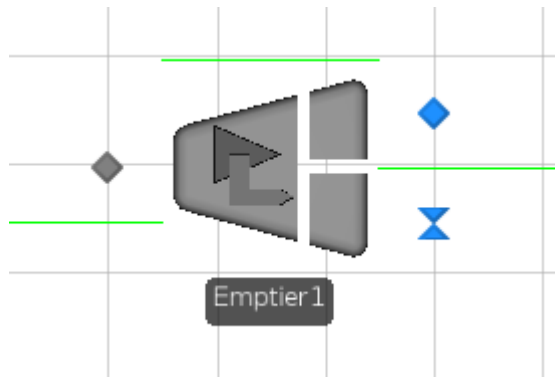
We have added pre-defined multiple animation symbols for many of the objects within the Standard and Flow Libraries. When you place an object, such as a Server, you will now see that there are multiple symbol values associated with that object, each with a distinguishing color.

For example, with a Server, there are 5 states including Starved, Processing, Blocked, Failed and Offshift. The default symbol value (0) is grey, but the Server automatically changes to green when the symbol value is 1 (Processing), 2 (Blocked), 3 (Failed) and 4 (OffShift). This is based on *the Current Symbol Index* for the Server, which is 'Server.ResourceState' by default.



If you resize the first object in the list, Simio will automatically resize all the corresponding objects. If you change the symbol from the default 'block' to something like a drill, for example, Simio will automatically change all associated pictures to the drill symbol, using the same color as the original (we do not change colors of new symbol pictures).

New Flow Library Object – Emptier



The Emptier object may be used to model an emptying process that removes flow product from container entities. The flow rate of the Emptier is regulated by its 'FlowOutput' flow node.

The Emptier is similar to the Server and Filler objects in that it has 3 stations, including the InputBuffer, Processing and OutputBuffer stations. Container entities may be emptied until they are completely empty or until a specific quantity has been removed. As with many objects in the Standard Library, the Emptier

object may have a work schedule, failures, state assignments and multiple add-on process triggers.

New SimBits

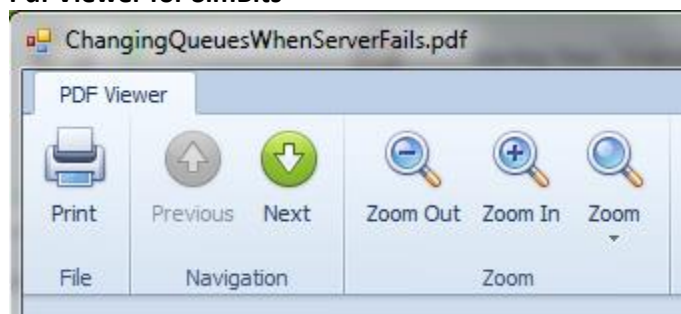
We have added two (2) new SimBits to our extensive SimBit library:

- **DisableFailureTimers.spfx** –This SimBit illustrates how to turn off Failures in a model that runs only until a given number of entities in the system has been processed. Typically, failure events will keep a simulation model running even though there may be no other events occurring in the system.
- **KeepQueueTimeForLast10Entities.spfx** –This SimBit demonstrates the use of storage queues to hold entity information while the actual entity is processing elsewhere. Copies of the most recent 10 entities in a queue are held in a storage queue for calculating a moving average type wait time statistics.

Evaluating Seize Request for Server, Combiner, Separator, Workstation, Filler, and Emptier

Like the Worker and Resource objects, there is now an 'Evaluating Seize Request' add-on process trigger available for Server, Combiner, Separator, Workstation, Filler, and Emptier. These processes can be used when the object is evaluating whether to accept or reject a request to seize capacity of the particular object. In the executed decision process, assigning a value of less than or equal to '0' to the executing token's ReturnValue state (Token.ReturnValue) indicates that the seize request is rejected. This feature allows the server-type object to do "smart" seizures like other resource objects without having to subclass them.

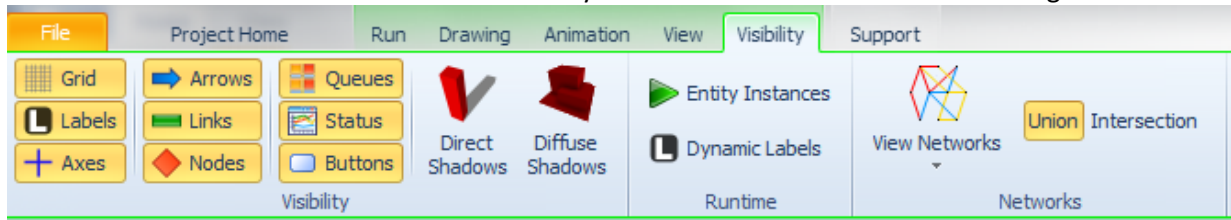
PDF Viewer for SimBits



We have added a new pdf viewer for displaying pdf file associated with any model files. This interface allows zooming in/out as well as printing. This allowed us to continue to show our pdf files on the desktop, as various windows operating systems were handling pdf viewers differently.

Entity Instances and Dynamic Labels on Visibility Ribbon

We have added two new features to the Visibility ribbon to control what is visible during runtime.

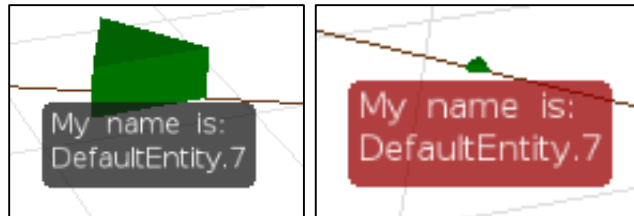


The **Entity Instances** icon will enable users to visibly turn on and off the Entity and Transporter instances placed in the Facility window during the simulation run. For old models, this icon is enabled and will show all instances of Entity and Transporter objects in the Facility window. For new models, this icon is initially disabled.

The **Dynamic Labels** icon works in conjunction with a new property on Entities and Transporters (which includes Worker / Vehicle) named *Dynamic Label Text* (in the Animation category of properties). The *Dynamic Label Text* is an expression which returns a string to show a floating label below a dynamic entity at runtime. The **Dynamic Labels** icon on the Visibility ribbon allows users to enable or disable these labels separately from the (static) Labels. This floating label is similar to the label that can be placed in the background, in that it is always “faces” the viewer and is always visible regardless of zoom level unless obstructed by another label or object. It is particularly useful when you have models covering a large scale and you want to locate an entity (for example see truck traffic on miles of roadway).

String.NewLine Function

The String.NewLine function will allow you to now put various strings together and a line is generated between the strings. This new string function is particularly useful in Write steps or with the Dynamic Labels described above. For example, within a Dynamic Label Text step, the expression "My name is:" + String.NewLine + Entity.Name will result in a label readable regardless of zoom level:



Settings Button on File Menu

We have added a Settings option on the main File menu which allows users to specify their graphics card for displaying 3D graphics (currently includes DirectX and Open GL). These application settings also include options to load (or suppress loading) the Standard Library and Flow Library, plus an option to specify multiple additional user libraries to load upon opening Simio. For example if you seldom use the Flow Library, but frequently use the Crane Library, you can reflect that here for automatic loading.

New Regulator Functions

OutputFlowReceivers.NumberItems - Returns the current number of entities receiving outflow from the regulator.

OutputFlowReceivers.FirstItem - Returns a reference to the first entity in the list of entities currently receiving outflow from the regulator.

OutputFlowReceivers.LastItem - Returns a reference to the last entity in the list of entities currently receiving outflow from the regulator.

OutputFlowReceivers.IndexOfItem(entity) - Returns a reference to the entity at a specified index position in the list of entities currently receiving outflow from the regulator.

OutputFlowReceivers.ItemAtIndex(index) - Returns the one-based index of a specified entity in the list of entities currently receiving outflow from the regulator. If the entity is not a current output flow receiver of the regulator then the value 0 is returned.

OutputFlowReceivers.Contains(entity) - Returns True (1) if the list of entities currently receiving outflow from the regulator includes the specified entity. Otherwise, the value False (0) is returned.

Table Property Functions

In previous software versions, the object/model could have a Table type property. However, there were some restrictions on when this property could be used. We now support of the following table property syntax:

TableProperty[row, column] - Returns a value from a row, column location of a table pointed to by a table property. NOTE: row and column must both resolve to integer values.

TableProperty.AvailableRowCount - Returns the number of rows for a table pointed to by a table property.

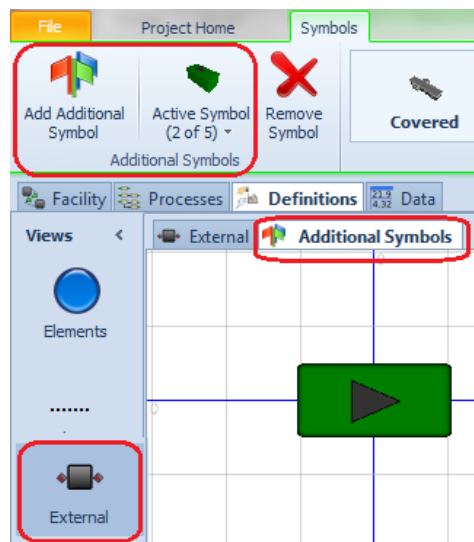
TableProperty.TimeIndexedRow - Returns the time indexed row of a table pointed to by a table property.

Some examples of using these table properties include the following:

TableProperty1[3, 4] returns the value at row 3, column 4 within the table referenced in TableProperty1.

MyFixed1.MyTableProp[3, 4] returns the value of row 3, column 4 within the table referenced in MyTableProp within some instance of an object named MyFixed1

Ability to Assign Additional Symbols for User Defined Objects through External Panel

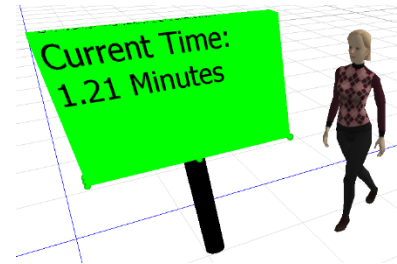


The multiple symbols on library objects feature previously described is available because of the new capability to assign additional symbols for an object's external view through the Definitions tab / External panel of the object.

Now, instead of just an External window, there is an Additional Symbols tab that will allow users to define the look, color and texture of multiple symbols for that object definition (not just the object instance when it is placed in a model).

Object Height on Status Objects

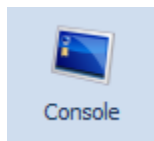
Object Height (on the Drawing ribbon) now applies to Status Label, Floor Label, Status Pie, Circular Gauge and Linear Gauge. If any of these have a non-zero height, they will be drawn as a box, with the same texture on the two largest sides. This is useful for creating a billboard or upright status information display as with the sign on a post illustrated here.



Support for Nautical Distances and Speeds

If you are modeling shipping systems you will appreciate that Simio now supports length units of Nautical Miles and speed units of Nautical Miles per Hour (commonly known as a “knot”). For you land lubbers, a nautical mile represents 1 minute of longitude at the equator and 1 knot is about 1.15 mph.

Dashboard Tab on Main Model/Object Changed to Console in Definitions Tab

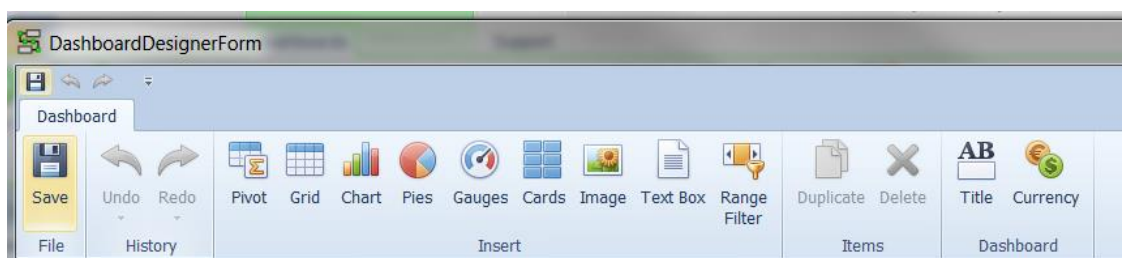


We have replaced the Dashboard tab within the main tab group (Facility, Processes, Definitions, Data, Results) with a Console panel option in the Definitions window. This Console window functions the same as the Dashboard window previously and includes the Animation ribbon for placing status displays in the window. Default Console windows are available for the Source, Server, Sink, Workstation, Combiner and Separator objects and can be accessed by right-clicking on the object and selecting the Open Console option.

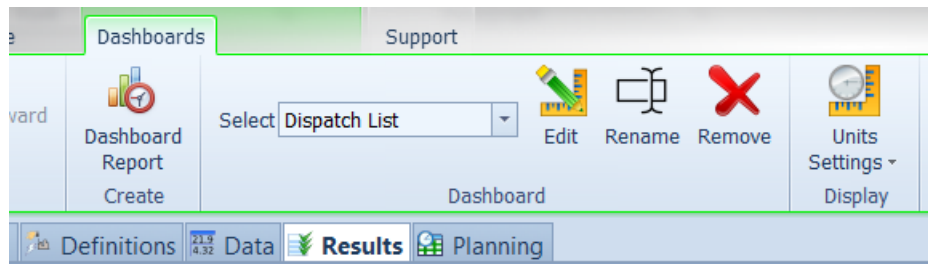
Similar to the previous Dashboard window, items within the Console window of the Definitions tab can be seen during the simulation run by tiling the windows such both the Facility window and Console panel of the Dashboard window area displayed.

Dashboard Reports

Dashboard *reports* (not to be confused with the recently renamed Dashboard *tab*, discussed above) allow you to display a dashboard based report of generated model data. A dashboard allows you to organize and present model information that is customizable and easy to read. Model data can be displayed in many graphical forms such as grids, charts and gauges. It also allows you to customize your dashboard by adding images, text boxes and filter ranges.

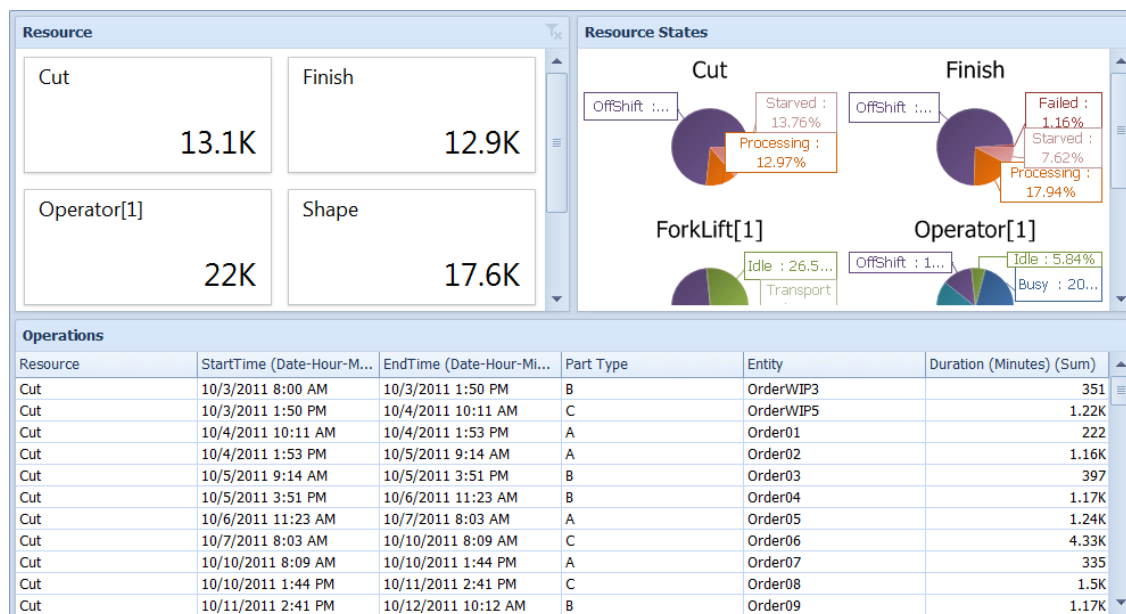


The major components of the Dashboard Report tool include creating a dashboard report and selecting a dashboard report to view, edit, rename or remove. There is also the option to set display settings such as time, length and distance, volume and mass and weight.



Depending on your license type, you can create and/or view dashboards that contain state and tally observations, user defined tables or other interactive logs:

- All editions can view previously created dashboards.
- Team or Academic edition is required to create new dashboards.
- Enterprise edition is required for full dashboard capability.



Gantt Visibility Option (Enterprise only)

There's a setting on each entity instance called "Gantt Visibility", an expression that is evaluated for each row in the Resource Usage Log, Transporter Usage Log, and Constraint log, to determine if this row should appear in the Entity Gantt chart.

Transporter Usage Log (Enterprise only)

The Transporter Usage Log now has "From Node" and "To Node" columns.

Simio Release 5 – Sprint 95 – October 8, 2013

In this release, we have enhanced our Flow Library to include two additional objects -- a Filler and a ContainerEntity, as discussed below. We've also added a number of SimBits to our extensive SimBit library, including some work flow (WF_*) type examples and several models displaying our new Flow Library objects.

This software continues to be a 'Beta' version of Simio 6 software and we appreciate any early feedback you may have.

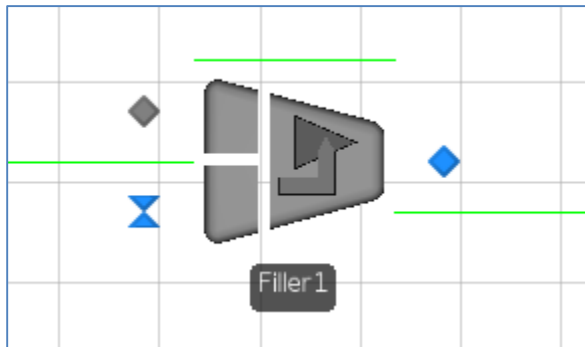
New Flow Library Object - ContainerEntity



The ContainerEntity object may be used to model a type of moveable container (e.g., barrels or totes) for carrying flow entities representing quantities of fluids or other mass.

ContainerEntities include a container element that may include initial contents within the container. Similar to a ModelEntity, the ContainerEntity may have travel and routing logic (initial desired speed, network, priority and sequence). Additional features, such as state assignments and add-on processes when the container is full and empty, are also available.

New Flow Library Object - Filler



The Filler object may be used to model a filling process that fills container entities with flow entities representing quantities of fluids or other mass. The filling flow rate of the Filler is regulated by its 'FlowInput' flow node. Container entities enter the Filler through its 'ContainerInput' basic node and exit the Filler from its 'Output' transfer node.

The Filler is similar to the Server in that it has 3 stations, including the InputBuffer, Processing and OutputBuffer stations. Container entities may be filled until they are full or to a specific quantity. As with many objects in the Standard Library, the Filler object may have a work schedule, failures, state assignments and multiple add-on process triggers.

New SimBits

We have added four new SimBits to our SimBit library:

- **ChangingQueuesWhenServerFails.spfx** –This SimBit shows the logic involved for searching an input queue of a server when the server fails to remove and redirect entities to another server for processing.
- **CheckingEntityAheadOnLink.spfx** –This SimBit uses the EntityAheadOnLink function on a link to evaluate and make decisions on entity link spacing.

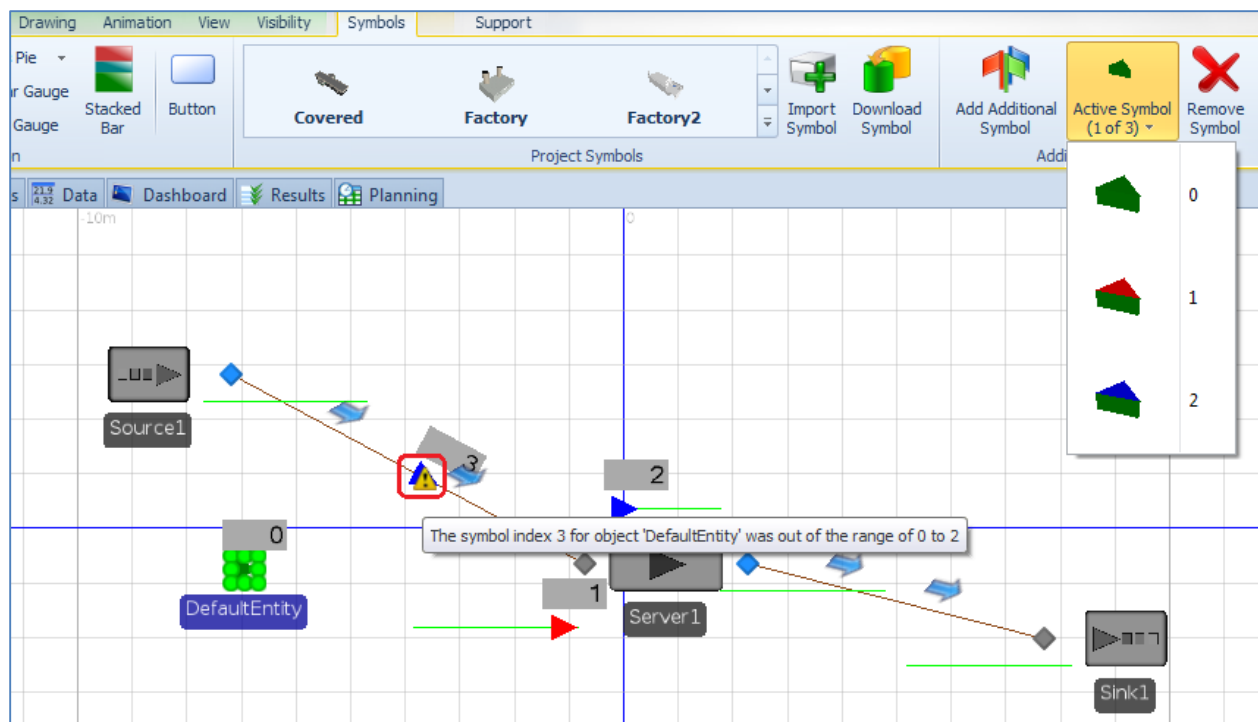
- **WF_AdditionalResource.spfx** –This SimBit illustrates a work flow environment where multiple entity types require a different number of resources for processing and functions are used for processing times.
- **WF_Authorization.spfx** – This SimBit illustrates the use of multiple sets of workers within a work flow environment such that various types of entities require workers with different sets of skills for processing.

We have also updated the **FlowConcepts** SimBit to include a new model:

- **SimpleFiller** – This model illustrates the use of the new Filler and ContainerEntity objects within the Flow Library. ContainerEntity objects enter the Filler as discrete entities, while flow from a FlowSource enters the Filler object as flow type entities. The containers are filled and a full ContainerEntity exits the Filler to a Sink.

Revised Warning for Animation Symbols – Out of Range

In prior Simio versions, if an object's Current Symbol Index value was 'out of range' based on the value assigned and symbols defined, a large warning would appear and temporarily halt the simulation model. We have now changed this behaviour such that if an object's animation symbol index is not within the defined range, an animated exclamation point, as shown below, is shown over the symbol with a warning displayed when the animation is paused and mouse hovered over the object. After several user's feedback on the original method, we have changed this warning to be less obtrusive and hopefully more suitable to all users.



Simio Release 5 – Sprint 94 – September 18, 2013

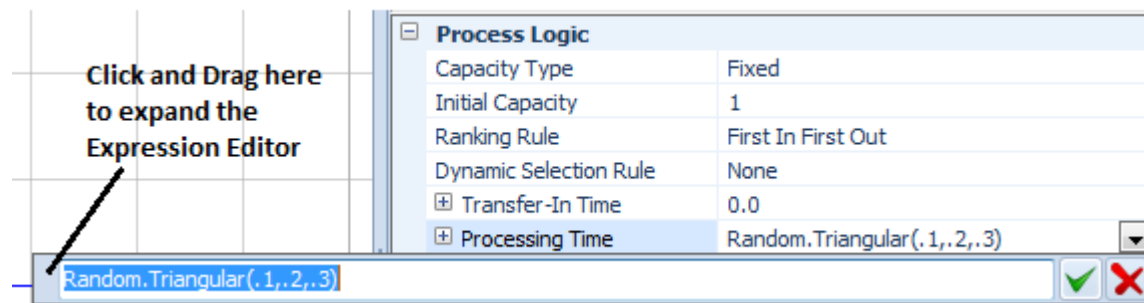
In this sprint release, we have added many miscellaneous features that have been requested by our users over time. We are also continuing to work toward Simio Release 6, which will be released later this fall. You may wish to consider this a 'Beta' version of that software, with much more to come. We appreciate any early feedback you may have.

Flow Library Enhancement

The FlowNode object (and associated nodes within the FlowSource and Tank objects) has been enhanced to include a *Split Allocation Rule* of 'Evenly If Possible' for splitting flow on the output flow from the node. When this rule is used, the flow regulator will distribute flow as evenly as possible to the node's outbound links while maximizing total outflow.

Expression Editor Resize

The Expression Editor can now be resized to allow for viewing/input of larger expressions. To resize the editor, simply click and drag the left portion of the expression editor and resize it. This size will remain for all expression editing until it is changed by the user.



Experiment Window Use of Control – Category Name

When model properties are defined, they may be specified into different categories, using the *Category Name* field on the property. We now utilize these category names when displaying the properties within an experiment as well.

Profile Reporting for Run Performance Evaluation

Within the Advanced Options button of the Run ribbon, there is a new option to Enable Run-Time Profiling. If this option is enabled, during interactive runs Simio will record execution counts and timing for various parts of the model. The results are summarized in the Profile window, which can be opened by selecting the Profile button in the windows area of the Project Home ribbon. This will allow users to evaluate where performance improvements to their model may be made.

Profile				
Item	% Time in Parent	% Time Overall	Count	Plot
▼ [Calendar Event]	99.98	99.9771	80702	
> Path.OnReachedEnd.FromPath [Transfer]	28.11	28.1014	9230	
[Simulation Engine]	19.75	19.7436	80702	
> Worker.OnVisitingNode.Worker [UnPark]	4.40	4.3983	1383	
> Server.OnEnteredOutputBuffer.ToOutputNode [Transfer]	4.04	4.0365	1473	
> OnRunInitialized.InitializationTable [Search]	2.81	2.8069	1	
> Bed.OnEnteredProcessing.SecondaryResource [Release]	2.37	2.3713	381	
> TransferNode.OnEntered.ToOutboundLink [Transfer]	2.11	2.1115	1206	
> Bed.OnEnteredProcessing.SecondaryResources [Seize]	1.91	1.9062	381	
> Server.OnEnteredProcessing.ToOutputNode [Transfer]	1.76	1.7613	755	
> Worker.OnVisitingNode.Worker [Release]	1.66	1.6601	501	

On Interval Process Option for Time-Indexed Tables

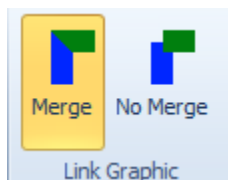
Within a Time-Indexed table (Advanced Options properties on the table), the optional *On Interval Process* is now available. When the index into the table is updated based on the time-interval specified, the On Interval Process is run at the start of each interval for any variable updating or calculations that a user may wish to make.

Gesture Support in 3D for Touch Screen Users

Pinch-to-zoom and two-finger-rotate gestures are now available within the 3D view, provided the user has a touch screen.

Merge/No Merge Options on Appearance Ribbon for Connecting Links

When a Node is highlighted, the Appearance ribbon now includes Merge and No Merge options for graphically showing the links that intersect at the node. Merge will smoothly connect the two paths, while No Merge will show a slight disconnect. This is primarily used when the paths coming into and out of the node include path decorators.



New Logs within the Results Tab

We now have three new logs that provide detailed information regarding state variables, tally statistics and material usage. For each of the State Statistic, Tally Statistic and Material elements, there is a new *Log Observations / Log Material Usage* option that can be specified as 'True' to enable logging.

The State Observation Log provides state variable information for those State Statistic elements that have logging specified as 'True'. When the value of a state variable or expression changes, that value is recorded in the log.

Views <

State Observation Log


Tally Observation Log


Material Usage Log


Time	Object Type	Object Name	Data Source	Category	Data Item	Value	Units	Rate	Units	Acceleration	Units
9/23/2013 12:00:00 AM	Model	Model	NumberInSystem_State	UserSpecified	StateValue	0		0		0	
9/23/2013 12:00:00 AM	Model	Model	NumberInSystem_State	UserSpecified	StateValue	1		0		0	
9/23/2013 12:00:00 AM	Model	Model	NumberInSystem_State	UserSpecified	StateValue	2		0		0	
9/23/2013 12:00:17 AM	Model	Model	NumberInSystem_State	UserSpecified	StateValue	1		0		0	
9/23/2013 12:00:23 AM	Model	Model	NumberInSystem_State	UserSpecified	StateValue	2		0		0	
9/23/2013 12:00:23 AM	Model	Model	NumberInSystem_State	UserSpecified	StateValue	3		0		0	
9/23/2013 12:00:31 AM	Model	Model	NumberInSystem_State	UserSpecified	StateValue	2		0		0	
9/23/2013 12:00:33 AM	Model	Model	NumberInSystem_State	UserSpecified	StateValue	3		0		0	
9/23/2013 12:00:33 AM	Model	Model	NumberInSystem_State	UserSpecified	StateValue	4		0		0	

The Tally Observation Log displays Tally Statistic observation values during the simulation run.

Views <


Pivot Grid


Reports


Logs

State Observation Log		Tally Observation Log		Material Usage Log				
	Time	Object Type	Object Name	Data Source	Category	Data Item	Value	Units
	9/23/2013 12:00:17 AM	Model	Model	FlowTime	UserSpecified	TallyValue	0.293581424629643	Minutes
	9/23/2013 12:00:31 AM	Model	Model	FlowTime	UserSpecified	TallyValue	0.527342431269617	Minutes
	9/23/2013 12:00:44 AM	Model	Model	FlowTime	UserSpecified	TallyValue	0.344947750602105	Minutes
	9/23/2013 12:00:56 AM	Model	Model	FlowTime	UserSpecified	TallyValue	0.552117402639884	Minutes
	9/23/2013 12:01:12 AM	Model	Model	FlowTime	UserSpecified	TallyValue	0.65412981339366	Minutes
	9/23/2013 12:01:21 AM	Model	Model	FlowTime	UserSpecified	TallyValue	0.791668676102233	Minutes
	9/23/2013 12:01:35 AM	Model	Model	FlowTime	UserSpecified	TallyValue	0.94792741865533	Minutes
	9/23/2013 12:01:48 AM	Model	Model	FlowTime	UserSpecified	TallyValue	1.16845109881255	Minutes
	9/23/2013 12:01:56 AM	Model	Model	FlowTime	UserSpecified	TallyValue	1.07556009563727	Minutes

The Material Usage Log displays information on materials that are produced and consumed during the simulation run.

Views

Pivot Grid

Reports

Logs

State Observation Log

Tally Observation Log

Material Usage Log

Time	Material	Entity	Quantity	Stock Level
1/1/2008 12:00:00 AM	Wood		30	30
1/1/2008 12:00:00 AM	Nails		60	60
1/1/2008 12:00:04 AM	Wood	DefaultEntity.11	-2	28
1/1/2008 12:00:04 AM	Nails	DefaultEntity.11	-8	52
1/1/2008 12:01:38 AM	Wood	DefaultEntity.12	-2	26
1/1/2008 12:01:38 AM	Nails	DefaultEntity.12	-8	44
1/1/2008 12:02:28 AM	Wood	DefaultEntity.13	-2	24
1/1/2008 12:02:28 AM	Nails	DefaultEntity.13	-8	36
1/1/2008 12:03:15 AM	Wood	DefaultEntity.14	-2	22

New Animations for Moving Objects (Walking People)

We have added a number of new libraries of 'animated' moving symbols. Prior to this sprint, we had animated males and females. Our animations now additionally include children, elderly persons, soldiers and toons (for fun ☺).

All groups

Library\People\Animated\Soldier



Soldier



Soldier1



Soldier2



Soldier3



Soldier4



Soldier5

Library\People\Animated\Toon\ToonBoy



ToonBoy



ToonBoy1



ToonBoy2



ToonBoy3



ToonBoy4



ToonBoy5



ToonBoy6

Simio Release 5 – Sprint 91 – July 16, 2013

In this sprint release, we have added several new features designed to help our beginning users, in addition to some new functions and SimBits for more advanced users. Within the Properties window of an object, you will now see a Show Commonly Used Properties checkbox (initially enabled) that will expose only those commonly used properties within an object. Of course, all of our features and flexibility are still there and can be accessed by unchecking this checkbox.

Show Commonly Used Properties – Option for New Users

At the top of the Properties window of an object, you will now see the checkbox *Show Commonly Used Properties*.



When this mode is enabled (the default), the display is limited to the key set of properties that defines the core behavior of each object. Many standard features such as failures, state assignments, secondary resource allocations, financials, and custom add-on process logic that are provided by the Standard Library objects are not displayed. This mode allows the beginner to focus on the key concepts for each object without any additional complexities.

When this option is disabled (unchecked), the full set of properties for each object is exposed. Note that this mode does not directly affect model behavior or results. It is just hiding/un-hiding properties but not changing property values. This mode is typically used when first learning Simio and then disabled once the basic concepts are mastered.

Elements Functions for Intelligent Objects

Every object now has an 'Elements' namespace that provides generic access to the collection of direct child elements contained within that object. These new Elements type functions may be used, for example, in a Find step to search all the direct child elements of an object to find a particular element and/or condition.

Elements.NumberItems- Returns the number of direct child elements contained within this parent object.

Elements.FirstItem - Returns a reference to the first element in the list of direct child elements contained within this parent object.

Elements.LastItem - Returns a reference to the last element in the list of direct child elements contained within this parent object.

Elements.ItemAtIndex(index) - Returns a reference to the element at a specified index position in the list of direct child elements contained within this parent object.

Elements.IndexOfItem(element) - Returns the one-based index of a specified element in the list of direct child elements contained within this parent object. If the element is not a direct child element of this parent then the value 0 is returned.

Elements.Contains(element) - Returns True (1) if the specified element is a direct child element of this parent object. Otherwise, the value False (0) is returned.

Enhanced Documentation

The e-book formerly named Introduction to Simio has new content including an introduction to simulation, simulation success skills, and an enhanced glossary. It has been renamed and is available on the Support ribbon.

New Videos

Two new video sets (Flexible Manufacturing Series and Learning Simio Lab Series) have been added to the existing Introduction to Simio course and all three are now directly available from the Support ribbon.

New SimBits

We have added five new SimBits to our SimBit library:

- **ChooseAlternateSequence.spfx** – This SimBit shows entities that have a typical main sequence of server steps, but due to inspection / rework process, some go through an alternate sequence and then return to the main sequence.
- **MergingConveyorsControlledByGate.spfx** – This SimBit uses a Resource object as a type of ‘gate’ to control the sequence of entities arriving from two merging conveyors.
- **VisitAllServersInAnyOrder.spfx** – This SimBit illustrates how a simple modification can be made to an object (Server) to permit more intelligent use, allowing selection of only objects that still need the service it provides. This is done by sub-classing a Server and adding a state variable for tracking the entities that have processed.
- **ExcelReadWrite.spfx** – This SimBit illustrates how to read and write using an Excel spreadsheet file.
- **DBReadWrite.spfx** – This SimBit illustrates how to read and write using a MySQL database.

Enhanced Button Animation

The Button on the Animation ribbon now has a *Text Scale* property to resize the text size on the button. This was requested by multiple users recently.

Model Properties

Model Properties has continued to be available via a left click menu, but you will see that it will now be displayed at other times, like when you click on the background of the facility view. The contents have now been better organized as well to prominently display only the properties of most interest. If you have not previously discovered it, when you create reference properties, they appear under a special “Controls” category in model properties. This makes a convenient interface for working interactively with a model.

Expression Editor

You will now find that the list of items on the Expression Editor pull down list is much shorter. We have now suppressed display of many items that are rarely used to make the frequently used ones more prominent.

Simio Release 5 – Sprint 89/90 – June 24, 2013

In this sprint release, we have added some subtle advanced features for object building and fixed a number of software issues that have been recently discovered. The frequency and content of our public releases has temporarily slowed while we concentrate our development resources on a major new feature. Watch for news on this coming soon.

Container Element ‘Empty’ and ‘Full’ Events

We have added the Empty and Full events to the Container element. These events are fired when a container is empty or full, respectively. This allows other objects to take some event-triggered action when a container completes a fill or empty operation.

Monitor Element Enhancement

We have enhanced the State Variable Name property of the Monitor element to allow the specific state variable that is to be monitored to be a member of an object or element reference state variable thus allowing the state variable monitored by an individual Monitor element to be dynamically changed. A Monitor element will dynamically ‘rewire’ to the state variable to be monitored each time the Monitor is enabled (i.e., the Enabled state changes from False to True).

For example, you can now monitor a state variable like `MyEntityReferenceState.SomeState` where `MyEntityReferenceState` is a state variable assigned to point to some entity. This is valid as long as the Monitor is always Disabled whenever the `MyEntityReferenceState` variable is set to Nothing. So, typical usage would be to disable Monitor, assign reference state to point at whatever entity you want the Monitor to monitor a state value on, and then enable the Monitor.

Custom Object Enhancement

You can now create a custom object that has an ‘Input’ external node that is inputting entities into a station, node, or container location that is specified as a state variable reference. So, you can create a new custom fixed object, go to the External panel view and add an External Node symbol, then specify the input location as a state variable. Then, dynamically assign that state variable during the run (depending on whatever logic) to have entities go into whatever location the state variable is currently assigned.

Consistency in NaN evaluations

NaN stands for “Not a Number” and is the value taken on by a numeric field when it does not have a valid numeric value. Examples include the average value of a statistic before any observations have been recorded or an expression like `Math.Sqrt(-1)`. We have improved the consistency of Largest/Smallest value selection rules and Minimize/Maximize expression searches when evaluating candidates with NaN values. We now ignore or discard those candidates that result in NaN (not a number) based on the expression specified.

Table Binding Enhancement

Shift-Click on the Remove Binding button within the Data window tables will ask the user if they would like to remove binding for all tables. This allows you to 'unbind' all tables with binding at one time. This is particularly useful when you are sharing a model with someone else and you wish to run the model using only the built-in data tables.

Simio Release 5 – Sprint 88 – May 14, 2013

In this sprint release, we've added a number of new user defined steps and elements to provide the capability to read and write to databases and Excel spreadsheets. In addition, we have added a few right-click options in different areas to quickly add breakpoints in the Processes window and to quickly add Links between recently placed nodes. Some of our other new features are for user flexibility, including allowing more options on the transfer related steps/objects and providing local culture arguments for string and datetime functions.

'Specific' and 'Outbound Link Name' Options on Transfer Step, BasicNode and TransferNode

We have added the 'Specific' option to the Transfer step, BasicNode and TransferNode for indicating the outbound link preference when transferring an entity out of a node. The Outbound Link Name property then allows the user to specify the link, either by entering the link directly or, more typically, by referencing a link reference state.

Examples of using this new feature may include using a global 'Switch' state variable that is being dynamically assigned to 'switch' outbound flow direction from the node according to some designed control system logic.

Alternatively, perhaps each entity is executing decision logic (using steps such as Find, Search, or Decide) to do dynamically determine which outbound link it wants to select, store that selected link in a state variable, and then transfer out from the node using that selected link by configuring the 'Specific' outbound link preference on the node.

Yet another example is perhaps a property or state variable on an entity that is specifying which outbound link to use. This property might be based on entity type, or perhaps a variable that has been dynamically assigned based on where the entity is in its processing sequence or if it is 'rework' etc.

String and DateTime Function Enhancements

For any of the below functions, if the useLocalCultureInfo argument is 'False' or unspecified, then the comparison will be culture insensitive.

String.Compare(string1, string2 [, useLocalCultureInfo]) - The useLocalCultureInfo argument indicates whether to use local culture-specific information to influence the string comparison.

String.CompareIgnoreCase(string1, string2 [, useLocalCultureInfo]) - The useLocalCultureInfo argument indicates whether to use local culture-specific information to influence the string comparison.

String.FromDateTime (dateTime [, formatString, useLocalCultureInfo]) - The datetime value may be converted to its equivalent string representation using a specified date and time format string. The formatString argument should contain either a single format specifier character (see .NET Standard Date and Time Format Strings) or a custom format pattern (see .NET Custom Date and Time Format Strings). The useLocalCultureInfo argument indicates whether to convert the datetime value to a string using local culture-specific format information.

String.ToDateTime (dateTime [, useLocalCultureInfo]) - The useLocalCultureInfo argument indicates whether to convert the string to a datetime value using local culture-specific format information.

String.FromReal (value [, formatString, useLocalCultureInfo]) - The numeric value may be converted to its equivalent string representation using a specified numeric format string. The formatString argument should contain either a valid standard numeric format specifier (see .NET Standard Numeric Format Strings) or any combination of custom numeric format specifiers (see .NET Custom Numeric Format Strings). The useLocalCultureInfo argument indicates whether to convert the numeric value to a string using local culture-specific format information.

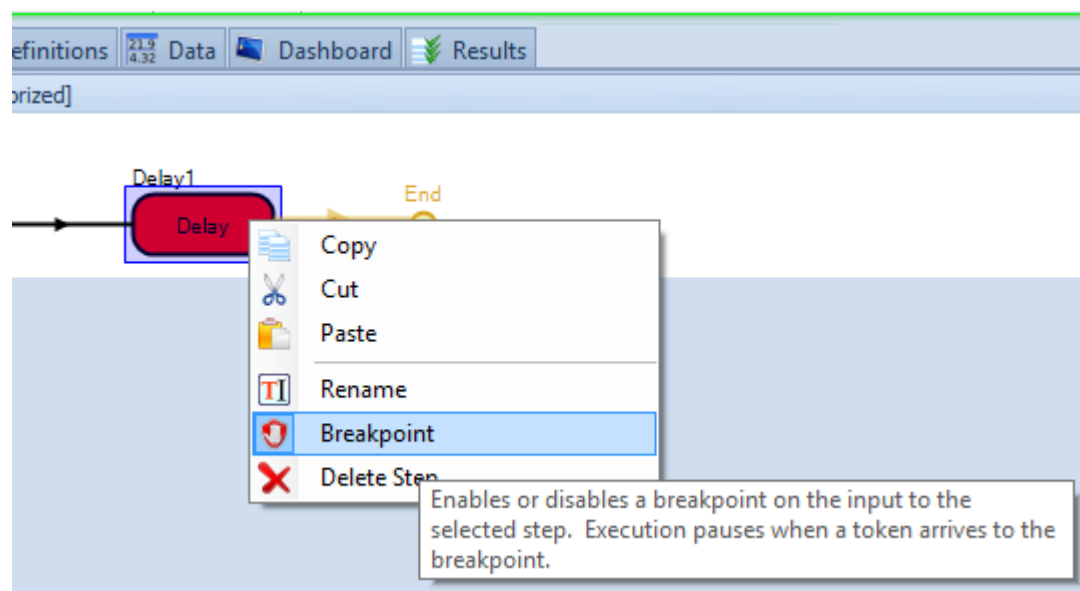
String.ToReal (string [, useLocalCultureInfo]) - The useLocalCultureInfo argument indicates whether to convert the string to a numeric value using local culture-specific format information.

DateTime.ToString (dateTime [, formatString, useLocalCultureInfo]) - The datetime value may be converted to its equivalent string representation using a specified date and time format string. The formatString argument should contain either a single format specifier character (see .NET Standard Date and Time Format Strings) or a custom format pattern (see .NET Custom Date and Time Format Strings). The useLocalCultureInfo argument indicates whether to convert the datetime value to a string using local culture-specific format information.

DateTime.FromString (string [, useLocalCultureInfo]) - The useLocalCultureInfo argument indicates whether to convert the string to a datetime value using local culture-specific format information.

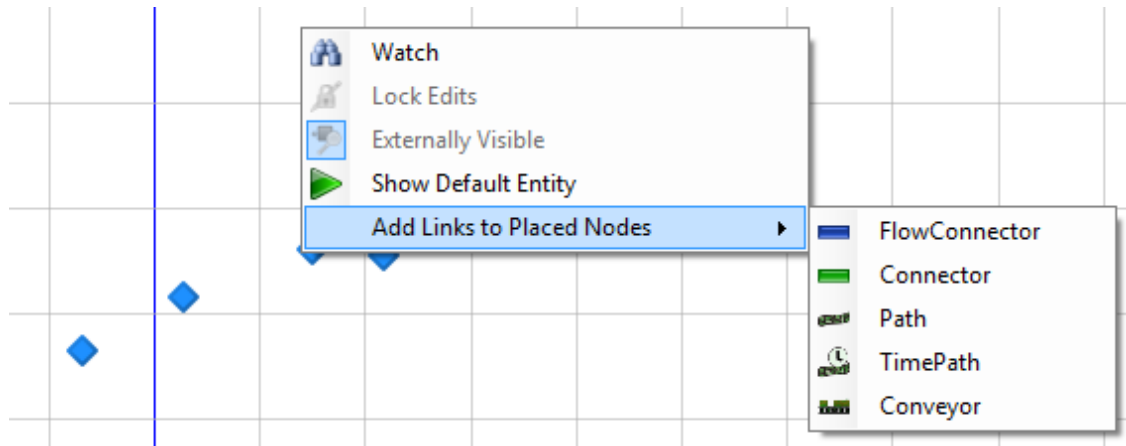
For Steps in Processes Window – New Breakpoint Option on Right-Click

Now, in addition to using the Breakpoint icon on either the Process or Run ribbons to set a breakpoint on a particular step, you can highlight the step and use the right-click menu, Breakpoint option, to turn on and off a breakpoint.



Ability to Add Links to Multiple Nodes Quickly

We have added a new feature to be able to quickly add links to a group of nodes that have been placed in the Facility window. Now, if the last two or more undo-able actions you did were placing nodes, you will see the 'Add Links to Placed Nodes' menu item in the right-click menu within the Facility window. This will allow you to create links of a specified type between the nodes that were just placed. For example, if you double click on the Standard Library object TransferNode to place multiple nodes and then click and place 5 nodes in the Facility window, you can right-click, select the new menu item (Add Links to Placed Nodes), select the link type (Path) and Simio will connect each of the first 4 nodes with a Path to the node that was subsequently placed.



After the links get created, they are all selected as well, so you can do bulk edits to them in the property grid.

Note that this only shows up if the last undoable action was placing a node. So if you place 3 nodes, then go to, say, the table view and create a table, then come back to the Facility window, you *will not* see this option, as the last undoable action you did was create a table.

Common Steps Panel Reorganized

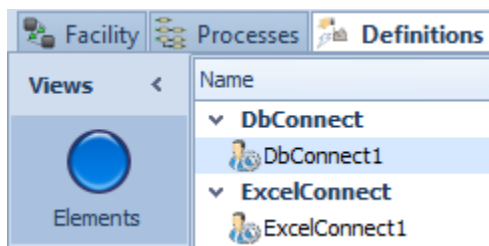
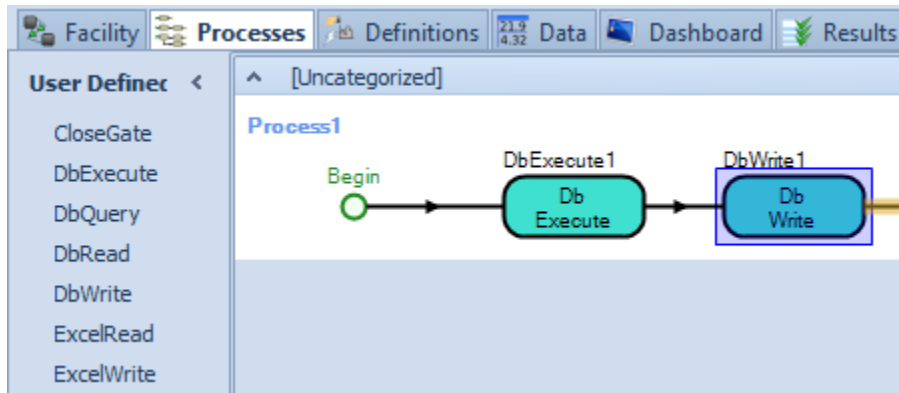
The steps within the Common Steps panel are now listed in alphabetical order instead of 'logical' type order. This was done based on several customer requests, as well as internal discussions on ease of use.

New User-Defined Steps and Elements for Database and Excel

We have added the six new user defined steps and two corresponding user defined elements for reading and writing to databases and Excel spreadsheets.

Within the User Defined panel in the Processes window, the database related steps include DbExecute, DbQuery, DbRead and DbWrite. The DbExecute step is designed to execute an SQL statement on a database. The DbQuery step may be used to query data from a database. The DbRead and DbWrite steps allow user to read from and write to a database. These steps require a corresponding element, the DbConnect user defined element, which specifies the database connection string and provider information.

Also within the User Defined panel of steps, the ExcelRead and ExcelWrite steps have been added to read values from and write values to an Excel spreadsheet. The corresponding element is the ExcelConnect, which specifies the Excel workbook information required.



Simio Release 5 – Sprint 87 – April 20, 2013

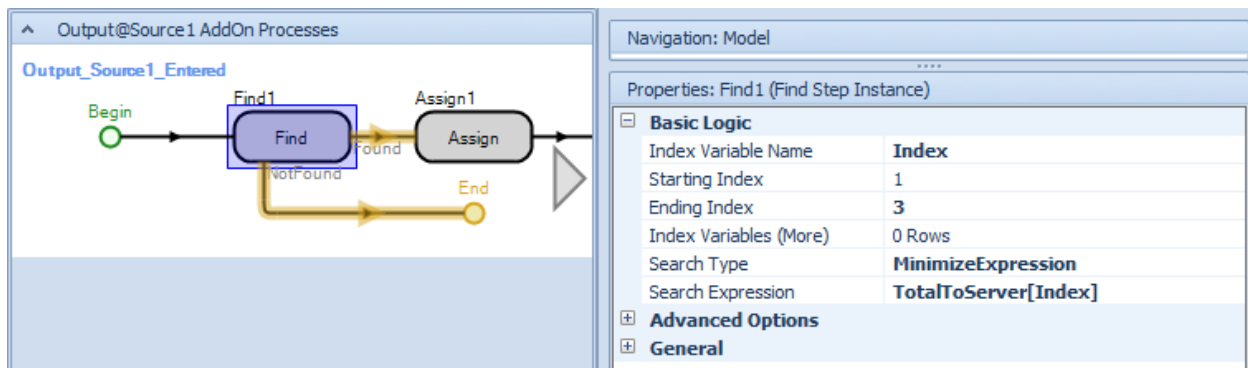
In this sprint release, we have added many user requested enhancements, including a much requested step to be able to search a state variable or array, the **Find step**. This allows users to evaluate values within a state array or matrices for a minimum or maximum value or one that meets a certain condition. It also allows for finding values within a table (similar to the Search step), except the Find step only returns the first found value.

Other user requested features include a new SimBit for flow, transparent option on floor labels, more flexibility in table and schedule referencing, and additional options on symbol import and database binding.

New Find Step

We have added a new Find step to the Processes window that may be used to search the value of an expression over a specified range of one or more indexed variables. The expression will typically involve array state variables (vectors or multi-dimensional arrays) or indexing related functions.

The Find step may be used to search a vector type state variable for a minimum or maximum value, for example, or to search the array for a given condition (values over the amount of X). This is a feature that many customers have asked for, as the Search step does not search for values within state variables.



New SimBits

We have added two new SimBits to our extensive SimBit library:

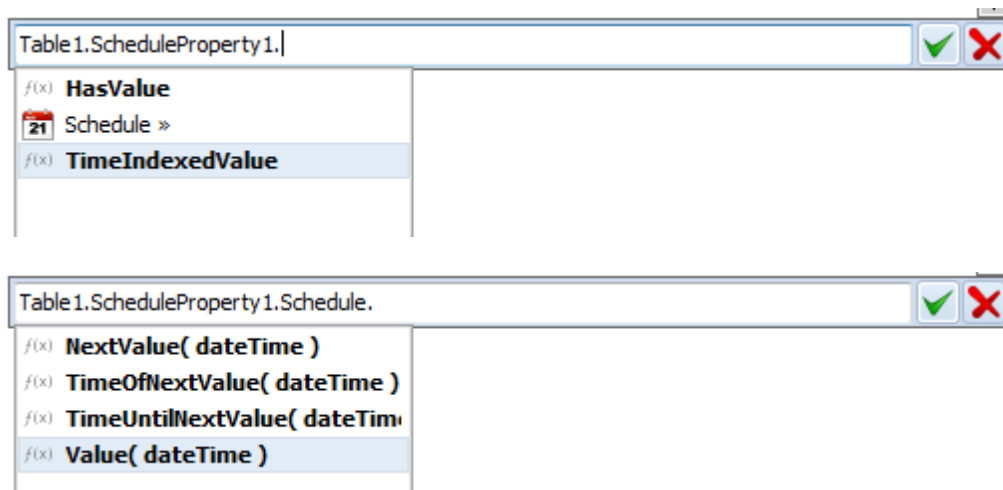
- **FindAMinimumStateValue.spfx** – This SimBit illustrates the use of the new Find step to search a state variable array for the minimum value.
- **PickUpDropOffFlow.spfx** – This SimBit demonstrates the Flow Library features in conjunction with an entity that has a container. The entity's container is filled at one location and then the entity travels to a destination where it empties its container.

Table Referencing Enhancement

We have added the ability to reference a data table entry as **TableName[row, column]** in an expression. This makes it easier, for example, to use Find step to search a table in the same manner as a variable matrix.

Schedule Referencing Enhancement

A Schedule property now provides access to the specified schedule data. This schedule property may be a property that is passing some schedule data into an embedded object's logic or may be a property within a table. Within the expression editor, you will now see for any schedule property access to the schedule functions.



Note that if you attempt to reference the schedule data stored in a schedule property, but no schedule name value has been specified for the property, then an 'Object or element reference not set to an instance of an object' error will be thrown by Simio. To avoid this error, the 'HasValue' function provided by any property (also a Sprint 87 new feature, seen below) may be used to do a conditional check that a property has a value before attempting to access the object or element reference stored in it.

Transparent Option on Floor Label

We have added the option for a transparent type background to floor labels. This was a user requested enhancement.

Symbol Rotation

We have added the option for rotating a symbol 180 degrees in the Symbol Import window. This was a user requested enhancement.

HasValue Function for Element and Object Properties

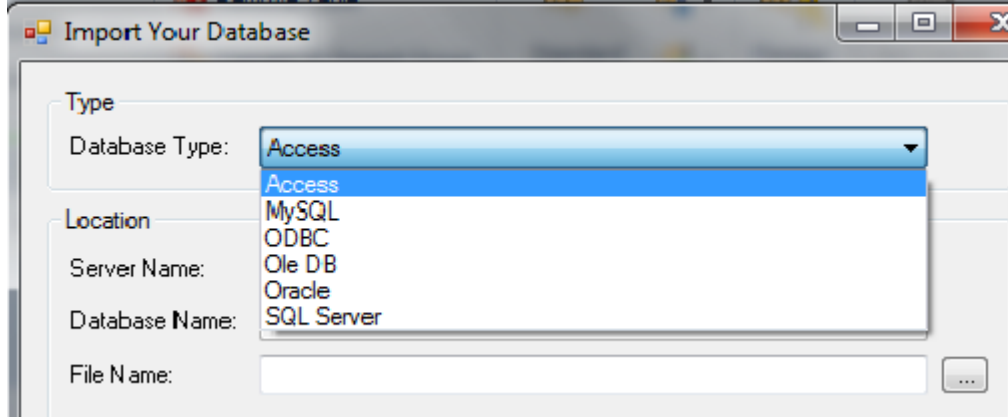
A new **HasValue** function has been added for all Element and Object type properties in the model. The function can be referenced as 'ElementPropertyName.HasValue' or 'ObjectPropertyName.HasValue'. This is useful for checking model logic whether an optional property has been specified a value or not.

The function HasValue returns True(1) if a value has been specified for the property. Otherwise the value False(0) is returned.

**** Repeat Group Note** – If the property is a repeat group, HasValue will return True(1) if one or more entries have been entered into the repeat group. Otherwise, the value False(0) is returned.

Database Binding Options – MySQL

Currently, there are many database type options when binding your table data to a database. We have now added the MySQL option to this list:



Performance Improvements

We have made some changes internally for run speed improvement that will impact some but not all models.

Simio Release 5 – Sprint 86 – April 4, 2013

In this sprint release, we have continued to enhance the Processes window steps to allow for specifying a particular object when performing a given step. These enhancements allow you to write process logic without requiring that the executing token is associated with the object that the step is acting upon, or requiring the process steps to have to be inside the 'Parent' object that they are acting upon.

We are sure you will enjoy our latest animation feature – walking people. In the software, we have provided a number of male and female animated symbols, located in Library\People\Animated in the Project Symbols area. These animated symbols have associated animations that allow them to walk, run, jump, swim and many other actions. Users can also import their own animations, if desired.

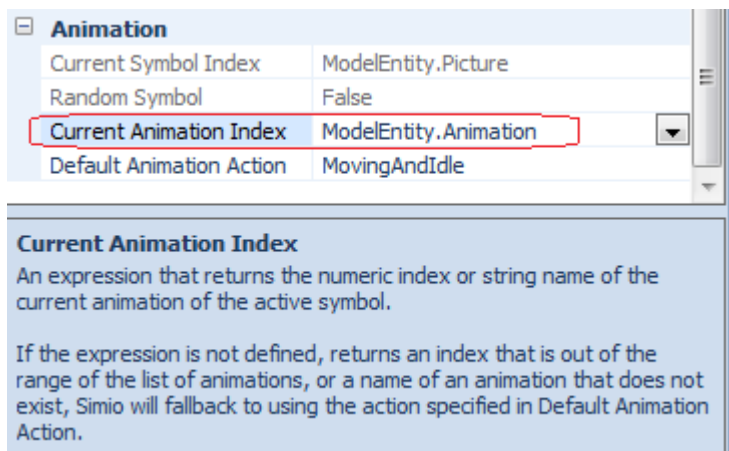
'Entity Type' and 'Entity Object' Enhancement on Batch, UnBatch, PlanVisit, SelectVisit, and Arrive Steps

The PlanVisit, SelectVisit, Batch, UnBatch and Arrive steps now support a 'SpecificObject' choice to specify the specific entity object to batch, unbatch, search visit requests, select destination of a visit request or notify visit request of arrival. All steps now include an *Entity Object* property for specific entity referencing and are no longer limited to just ParentObject or AssociatedObject choices for the *Object Type* property.

Animated Objects (Walking People)

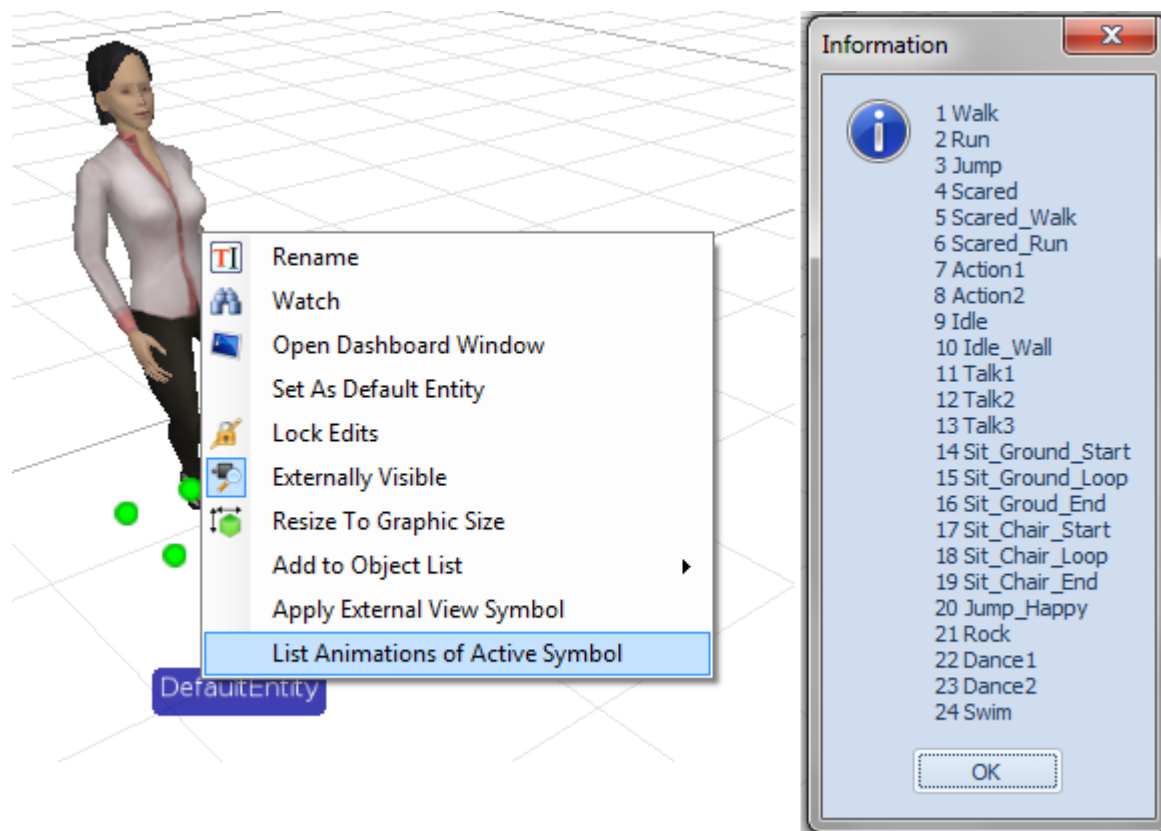
We have enhanced our animation support to include animations, which provide more realistic movement of entities through the system. Simio includes multiple male and female symbols within the Library\People\Animated area of the project symbols.

The ModelEntity within the Project window now has a string state named Animation, referred to within the model as ModelEntity.Animation. This string state is then used within the ModelEntity properties window as the default value for the Current Animation Index (see below).



Once an entity is placed in the Facility window, the default green arrow symbol can be replaced by an animated person (from the Library\People\Animated directory within the Project Symbols). Right-

clicking on an animated object will allow you to select *List Animations of Active Symbol*, which in turn will open the list of animations for that particular object, as shown below.



The animations may be referenced by either numeric index (in the case above, 1 through 24) or by string value (i.e., “Walk”). When using the `ModelEntity.Animation` string state, the string values above should be used. For example, in an Assign step upon arrival within a Source, the `ModelEntity.Animation` may be set directly to “Run” or could be assigned to be `Random.Discrete(“Run”, .25, “Dance1”, .5, and “Swim”, 1)`. In that case, 25% of the entities would run, 25% would dance and the other 50% would swim.

Alternatively, the `Current Animation Index` property itself may reference either the string or number directly (instead of using the default value of `ModelEntity.Animation`).

The entity’s speed, `ModelEntity.DesiredSpeed`, will also impact the entity movement (i.e., a slower moving entity will appear to be walking, while a faster moving entity be running).

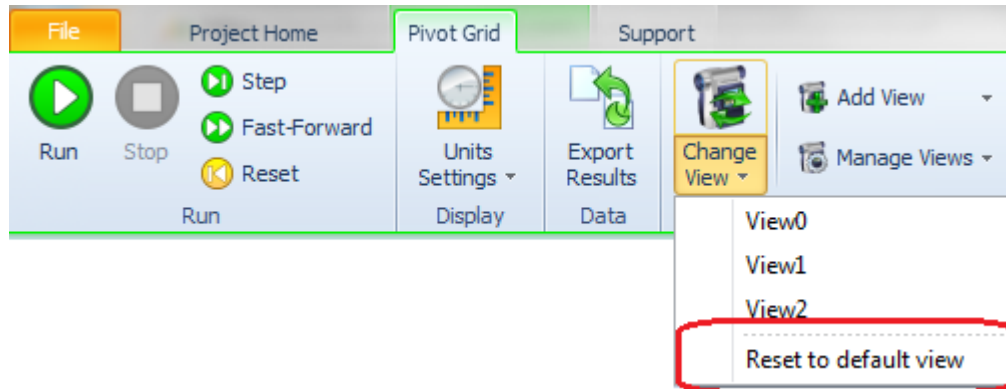
New Animation SimBit

We have added a new SimBit to our SimBit library:

- **AnimatedPeople.spfx** – This SimBit includes a demonstration of various ways to animate moving people (entities) in the system.

New 'Reset to default view' in Pivot Grid Window

In addition to being able to add, manage and change views within the Pivot Grid window, you may now also reset the pivot grid table to the original default view.



Simio Release 5 – Sprint 84/85 – March 14, 2013

In this sprint release, we have added many new functions and enhanced multiple steps to make the product more flexible. See the note below for examples of using these new features.

Additionally, several items have been added to the Simio Enterprise Edition to provide additional information to the end user on transporters and schedules, including new logs and Gantt changes.

Finally, we've started work on animated people and this sprint will provide a glimpse at that capability. Documentation, help and examples of this functionality will be provided within the next sprint, but check out the ThemePark example for a quick look!

General Note about New Step Enhancements and Functions

Enhancements made to many of the steps (as noted below) are simply to allow you to write process logic without having to be required that the executing token is associated with the object that the step is acting upon, or requiring the process steps to have to be inside the 'Parent' object that they are acting upon. This flexibility may be particularly useful to object builders. Many of these features, for example, are used to simplify modeling within the Crane library.

'Controller' process logic external to a vehicle, for example, can now use Pickup/Dropoff steps to command the vehicle to pick up or drop off entities. Or, you could model a train where all the pickup/dropoff logic is inside a single 'Locomotive' object, and the locomotive's logic is going to direct multiple 'Railcar' transporters pulled by the locomotive to pick up or drop off cargo. Previously, you were forced to put the Pickup and Dropoff steps inside each individual railcar.

Many of the new functions have been added for flexibility as well. You can now use the various schedules functions to determine when a particular resource will be changing shifts and make routing decisions accordingly, for example.

New Schedules Functions

Value(dateTime)- Returns the schedule's value for the specified numeric datetime (simulation time).

NextValue(dateTime) - Returns the schedule's next value if at the specified numeric datetime (simulation time).

TimeOfNextValue(dateTime) - Returns the simulation time (in hours) of the schedule's next value change if at the specified numeric datetime (simulation time).

TimeUntilNextValue(dateTime) - Returns the time duration remaining (in hours) until the schedule's next value change if at the specified numeric datetime (simulation time).

New Resource Object Functions

Capacity.Next - Returns the next scheduled capacity of this resource. *Note: If the resource is not following a work schedule then this function simply returns the current capacity since any future capacity changes are unknown.

Capacity.TimeOfLastChange - Returns the simulation time (in hours) that the capacity of this resource most recently changed.

Capacity.TimeOfNextChange - Returns the simulation time (in hours) of the next scheduled capacity change of this resource. *Note: If the resource is not following a work schedule then this function simply returns infinity since the timing of any future capacity changes is unknown.

Capacity.TimeSinceLastChange - Returns the elapsed time duration (in hours) since the most recent capacity of this resource.

Capacity.TimeUntilNextChange - Returns the time duration remaining (in hours) until the next scheduled capacity change of this resource. *Note: If the resource is not following a work schedule then this function simply returns infinity since the timing of any future capacity changes is unknown.

ResourceOwners.Contains(owner) – Returns True (1) if the list of owners that have currently seized capacity units of this resource includes the specified owner. Otherwise the value False (0) is returned.

New Token Function

AssociatedObject - Returns the object that this token is associated with.

New Entity Functions

NetworkDistanceTo.NextEntityAheadOnLink- If the entity object's leading edge is on a link, then this function returns the distance from the entity's leading edge to the trailing edge of the next entity ahead on the same link. If there is no entity ahead then the value Infinity is returned.

NetworkDistanceTo.NextEntityBehindOnLink - If the entity object's trailing edge is on a link, then this function returns the distance from the entity's trailing edge to the leading edge of the next entity behind on the same link. If there is no entity behind then the value Infinity is returned.

Population.Contains(entity) – Returns True (1) if the population contains the specified entity. Otherwise the value False (0) is returned.

Sequence.DestinationNodes.Contains(node) – Returns True (1) if the entity's assigned sequence contains the specified destination node. Otherwise the value False (0) is returned.

New Queue State Function

Contains(object) – Returns True (1) if the queue contains the specified object. Otherwise the value False (0) is returned.

New Node Functions

InboundLinks.Contains(link) – Returns True (1) if the collection of drawn inbound links into this node contains the specified link. Otherwise the value False (0) is returned.

OutboundLinks.Contains(link) – Returns True (1) if the collection of drawn outbound links from this node contains the specified link. Otherwise the value False (0) is returned.

New Network Element Function

Link.Contains(link) – Returns True (1) if the network's collection of link members contains the specified link. Otherwise the value False (0) is returned.

New Intelligent Object Function

SeizedResources.Contains(resource) – Returns True (1) if the list of resources currently seized and owned by this object includes the specified resource. Otherwise the value False (0) is returned.

‘Resource Type’ Enhancement on Allocate Step

The Allocate steps now has an additional ‘SpecificObject’ option for the ‘Resource Type’ property that allows the owner that the step’s action is in behalf of to be specified as a specific object reference. Before, the owner always had to be either the parent object containing the step or the associated object of the token executing the step. Similar changes were made to many other steps in the previous sprint to provide more flexibility in process logic design.

‘Entity Object’ Enhancement on Destroy, Route, SetNetwork, SetNode, Park and Unpark Steps

The Destroy, Route, SetNetwork, SetNode, Park and Unpark steps now support a ‘SpecificObject’ choice to specify the specific entity object to destroy, route, set the network, set the node, park or unpark. All steps now include an *Entity Object* property for specific entity referencing and are no longer limited to just ParentObject or AssociatedObject choices for the *Object Type* property.

‘Transporter Type’ and ‘Transporter Object’ Enhancement on Pickup, Dropoff and SelectDropoff Steps

The Pickup, Dropoff and SelectDropoff steps have been enhanced to allow the transporter to be more flexibly specified. Previously, these steps could only be used in process logic of the parent transporter to which the step actions pertained. The *Transporter Type* property allows the user to select the object associated with the executing token, the parent object or a specific object reference. The *Transporter Object* property is visible when the *Transporter Type* is ‘SpecificObject’ and is the specific object to perform the pickup, dropoff or select dropoff.

‘Object’ Enhancement on Insert and Remove Steps

The Insert and Remove steps now support a ‘SpecificObject’ choice to specify the specific entity object to insert into or remove from a queue. These steps now include an *Object* property for specific object referencing and are no longer limited to just ParentObject or AssociatedObject choices for the *Object Type* property.

Transfer Step Enhancement

When transferring from the current node to an outbound link, there are now an *Outbound Link Preference* and *Outbound Link Rule* properties which can override the default settings for those property values that are defined on the node.

Model Warning Added

We now give a warning when the total number of tokens in the system is more than 100,000. We also provide some course detail as to where the problem (if there is one) might be in the model. This will help prevent memory problems when a user may have an undetected logic issue.

Enhance State Arrays/Matrices Dimension Types

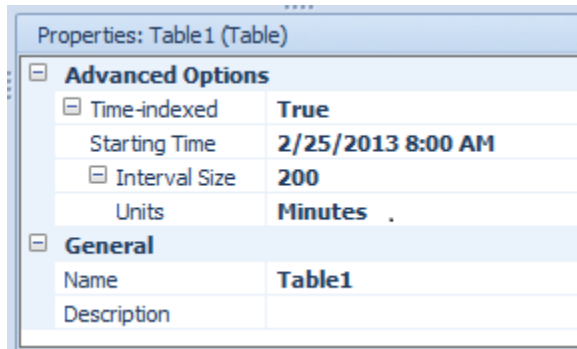
Within states, such as real, integer, object references, element references (all states but Lists), the *Dimension Type* property has been enhanced to include 3-dimensional up to 10-dimensional array options. This capability was previously within Simio, however, not as clearly stated.

Time-Indexed Table Properties

A data table may now be specified as Time-indexed, based on the properties of the table, as shown below. When the Time-indexed property is 'True', you may specify the Starting Time and Interval Size. Within a time-indexed table, you can then have a column(s) of values within the table and retrieve a single value from the column, not by direct indexing, but indirectly, based on the current simulation time. Two additional table functions have been added, including:

TableName.PropertyName.TimeIndexedValue – returns the value of the property in the row that corresponds to the current simulation time.

TableName.TimeIndexedRow – returns the row index that corresponds to the current simulation time.




Properties: Table1 (Table)	
Advanced Options	
Time-indexed	True
Starting Time	2/25/2013 8:00 AM
Interval Size	200
Units	Minutes
General	
Name	Table1
Description	

For example, let's say you had a column of values (IntegerColumn1) within Table1 that were 10, 20, 30, 40 and 50, using the above *Interval Size*. Then, from time 0 – 200 minutes, the `TableName1.IntegerColumn1.TimeIndexedValue` would return a value of 10. For the next 200 simulation minutes, that same function would return the value of 20 and so on.

New CSV Separator for Projects

Within any given project, in Project Properties, there is now a field to indicate the separator character to use when writing to a .csv file.

Properties: MySimioProject (Project)

Data	
CSV Separator	
General	
Project Name	MySimioProject
Icon	 (none)

CSV Separator
The separator character used when exporting data to a CSV file. If blank, Simio will use the "List Separator" character specified by the computer's current regional settings.

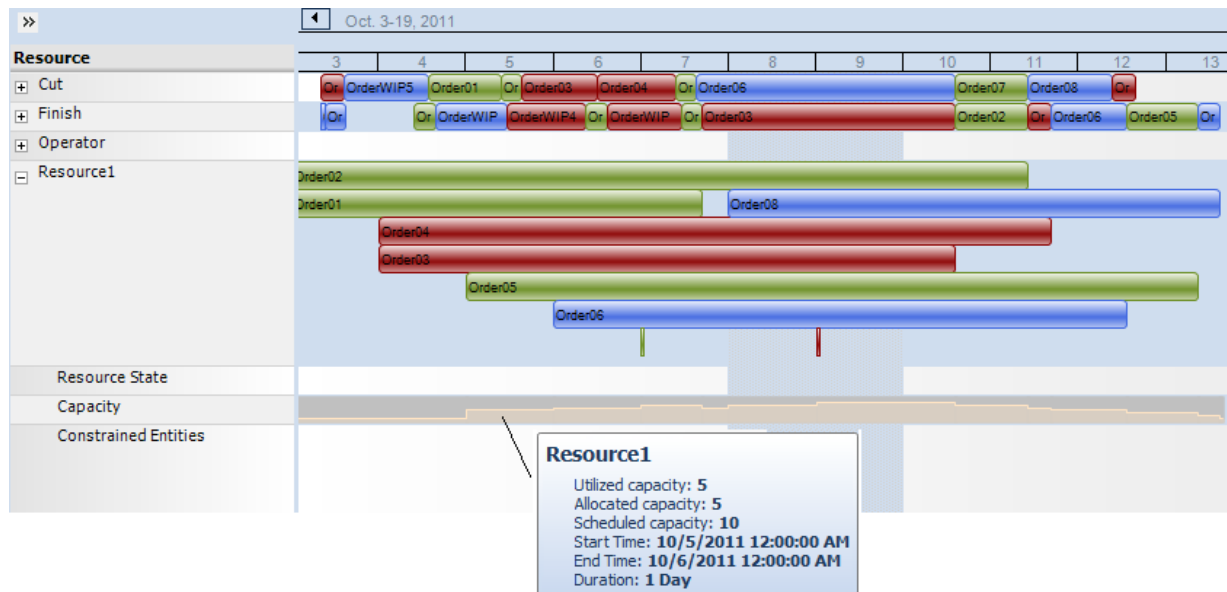
You can enter a specific character, but the preferred thing to do is to leave it at its default value of blank, and Simio will use the "List Separator" character defined in Windows' regional settings.

Simio Enterprise Edition – New Resource Capacity Log and Capacity row on Multiple Capacity Resources in Resource Plan Gantt

Within the Simio Enterprise Edition, we have added a Resource Capacity Log to the Logs section of reports. This log displays all of the capacity changes for the resources during the simulation run. This is especially useful when resource capacity is based on a work schedule. The capacity of the resource is displayed, along with the number that has been 'Allocated' (seized), as well as the number 'Utilized'. The utilized value may differ from the allocated value when a failure occurs or when a resource is waiting for material or a secondary resource before processing.

Resource Usage Log		Resource State Log		Resource Capacity Log		Constraint Log	Transporter Usage Log
Resource	Scheduled	Allocated	Utilized	Start Time	End Time	Duration (Minutes)	
► Shape	0	0	0	10/3/2011 12:00:00 AM	10/3/2011 8:00:00 AM	480	
Weld	0	0	0	10/3/2011 12:00:00 AM	10/3/2011 8:00:00 AM	480	
Cut	0	0	0	10/3/2011 12:00:00 AM	10/3/2011 8:00:00 AM	480	
Finish	0	0	0	10/3/2011 12:00:00 AM	10/3/2011 8:00:00 AM	480	
Shape	1	0	0	10/3/2011 8:00:00 AM	10/3/2011 12:00:00 PM	240	
Weld	1	1	0	10/3/2011 8:00:00 AM	10/3/2011 8:00:02 AM	0.0406667	
Operator[1]	1	1	1	10/3/2011 8:00:00 AM	10/19/2011 9:38:29 AM	23138.5	
Cut	1	1	1	10/3/2011 8:00:00 AM	10/3/2011 12:00:00 PM	240	
Finish	1	1	1	10/3/2011 8:00:00 AM	10/3/2011 12:00:00 PM	240	
Weld	1	1	1	10/3/2011 8:00:02 AM	10/3/2011 12:00:00 PM	239.959	

Additionally, within the Resource Plan Gantt, when a resource has a capacity greater than one, a new row named Capacity will be displayed and a usage type gantt shown for the capacity allocated for that resource over time. In the below example, Resource1 has a capacity of 10 (gray area). The tan area shows the number of resources allocated over time.



Simio Enterprise Edition – New Transporter Usage Log

Also within Simio Enterprise, we have added a Transporter Usage Log to the Logs reports. This log displays the usage of any transporters (used for transporting, not being Seized). The start time is recorded once the transporter has arrived and loaded the entity and ends once the entity is done being unloaded at its destination. This log corresponds to the transporter's information within the Resource Plan Gantt chart as well.

Resource Usage Log	Resource State Log	Resource Capacity Log	Constraint Log	Transporter Usage Log
Transporter	Rider	Start Time	End Time	Duration (Minutes)
Vehicle1[1]	OrderWIP6	10/3/2011 9:17:00 AM	10/3/2011 9:22:09 AM	5.1665
Vehicle1[1]	OrderWIP7	10/3/2011 3:05:09 PM	10/3/2011 3:10:19 PM	5.1665
Vehicle1[1]	OrderWIP5	10/5/2011 9:31:37 AM	10/5/2011 9:36:47 AM	5.1665
Vehicle1[1]	OrderWIP1	10/5/2011 3:19:37 PM	10/5/2011 3:24:47 PM	5.1665
Vehicle1[1]	OrderWIP4	10/6/2011 11:41:13 AM	10/6/2011 11:46:23 AM	5.1665
Vehicle1[1]	OrderWIP2	10/7/2011 8:29:13 AM	10/7/2011 8:34:23 AM	5.1665
Vehicle1[1]	OrderWIP3	10/7/2011 2:50:49 PM	10/7/2011 2:55:59 PM	5.1665
Vehicle1[1]	Order06	10/11/2011 10:10:01 AM	10/11/2011 10:15:11 AM	5.1665

Simio Enterprise Edition – New Gantt Chart Visible/Editable Properties and Gantt Visibility Options

Several smaller items have been enhanced within Simio Enterprise edition including the addition of Gantt Chart *Visible* and *Editable* properties within a data table. This will allow a particular column(s) to be displayed on the Resource Plan Gantt when the table is a Time-indexed type table.

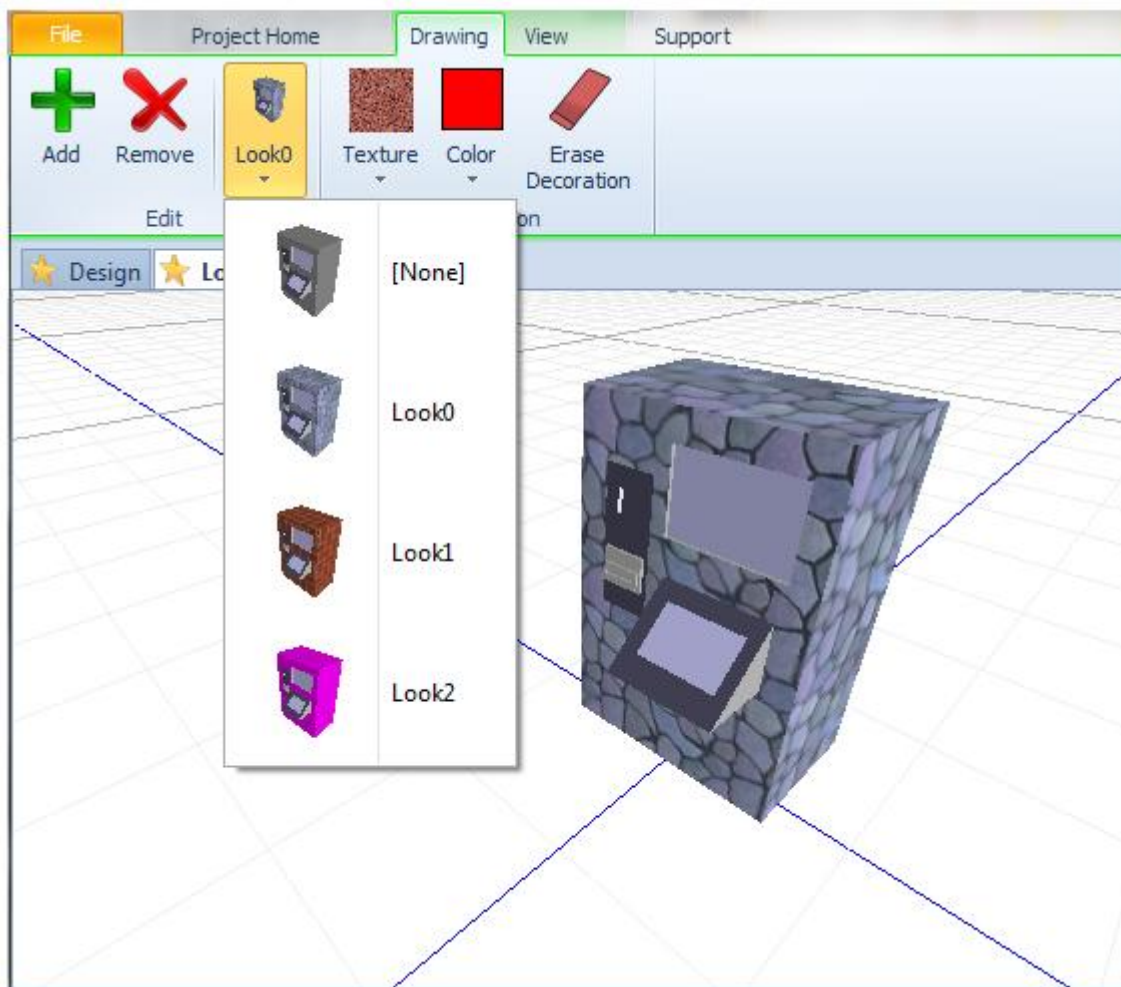
Additionally, various rows may now be toggled on/off within the Resource Plan Gantt by selecting the Gantt ribbon. Rows, such as Resource States, Exceptions and Constraints, can be turned on/off for more specific focus within a particular Gantt chart.

Animated Walking People

Simio now includes walking men and women within the symbol library. When selecting a symbol for an entity, for example, you can now select from multiple types of animated people, found within project symbols in the Library\Animated\People area. Full documentation/help/examples will be introduced in our next sprint. In the meantime, check out the ThemePark example for a demonstration of our latest animation feature!

Additional Symbol 'Look'

Within the Symbol window, we've added a new 'Look' (the previous symbol definition window is now the 'Design' of the Symbol window). For those situations where a single symbol 3d geometry may have several different textures or colors applied to it, the 'Look' allows users to define each of those. These new looks are then put into the library of symbols to be selected for any given project. This saves on both disk and video memory compared to having X different symbols with the same geometry.



Simio Release 5 – Sprint 82/83 – January 31, 2013

This is our first update after the release of Simio sprint 5 and we think users will be excited about these new flexibility enhancements. We've added a breakpoints window which will allow much greater flexibility in simulation debugging using breakpoints. Multiple functions for nodes and networks, along with increased capability within the Search step will allow users more control over network and link-based logic decisions in their models. Another major enhancement is the addition to many of our steps to allow entity reference states to be used when seizing, releasing, transferring, moving, producing, consuming and searching. And for those users with Simio Enterprise edition or those using the Flow Library, there are several enhancements as well.

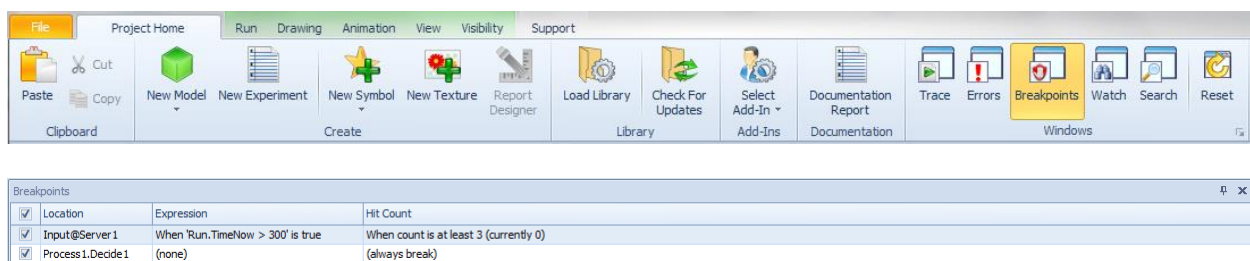
New Breakpoints Icon in Project Home Ribbon and Breakpoints Window

A new Breakpoints icon has been added to Windows section of the Project Home window. This opens a Breakpoint window that displays the various breakpoints that have been set by the user in either the Facility or Processes window.



This window also provides additional functionality for turning on / off breakpoints with a checkbox, as well as setting an expression and/or count that must be met for the breakpoint to execute. Expressions may involve simulation time, specific object information (number in a queue), functions, etc. Hit count can be specified as Always, Equal To (count) or At Least (count). The count is not evaluated unless the expression (if any) evaluates to true.

For example, in the screenshot below, the breakpoint on Input@Server1 will not be executed until after 300 time units into the simulation. Then, once the expression is true, the hit count is evaluated. The breakpoint will activate only when the location has been passed at least 3 times (3 or more). A current count is also displayed.



New Entity Function

Population.Name – Returns the string name of the entity's population instance. This is useful when working with just an object instance (e.g., ForkLift5) rather than the object definition (e.g., ForkLift).

New Node Functions

InboundLinks.NumberItems - Returns the number of drawn inbound links into this node.

InboundLinks.FirstItem - Returns a reference to the first link in the collection of drawn inbound links into this node.

InboundLinks.LastItem - Returns a reference to the last link in the collection of drawn inbound links into this node.

InboundLinks.ItemAtIndex(index) - Returns a reference to the link at a specified index position in the collection of drawn inbound links into this node.

InboundLinks.IndexOfItem(link) - Returns the one-based index of a specified link in the collection of drawn inbound links into this node. If the link is not an inbound link into this node then the value 0 is returned.

OutboundLinks.NumberItems - Returns the number of drawn outbound links into this node.

OutboundLinks.FirstItem - Returns a reference to the first link in the collection of drawn outbound links into this node..

OutboundLinks.LastItem - Returns a reference to the last link in the collection of drawn outbound links into this node.

OutboundLinks.ItemAtIndex(index) - Returns a reference to the link at a specified index position in the collection of drawn outbound links into this node.

OutboundLinks.IndexOfItem(link)- Returns the one-based index of a specified link in the collection of drawn outbound links into this node. If the link is not an outbound link into this node then the value 0 is returned.

New Network Element Functions

Links.NumberItems – Returns the number of links that are members of this network.

Links.FirstItem – Returns a reference to the first link in the collection of links that are members of this network.

Links.LastItem – Returns a reference to the last link in the collection of links that are members of this network.

Links.ItemAtIndex(index) – Returns a reference to the link at a specified index position in the collection of links that are members of this network.

Links.IndexOfItem(link) – Returns the one-based index of a specified link in the collection of links that are members of this network. If the link is not a member of this network then the value 0 is returned.

New Functions for any Resource object:

ResourceOwners – The objects that currently own (have seized) capacity units of this resource.

ResourceOwners.NumberItems – Returns the number of objects that currently own (have seized) capacity units of this resource.

ResourceOwners.FirstItem – Returns a reference to the first owner in the list of owners that have currently seized capacity units of this resource.

ResourceOwnters.LastItem – Returns a reference to last owner in the list of owners that have currently seized capacity units of this resource.

ResourceOwners.ItemAtIndex(index) – Returns a reference to the owner at a specified index position in the list of owners that have currently seized capacity units of this resource.

ResourceOwners.IndexOfItem(owner) – Returns the one-based index of the first occurrence of a specified owner in the list of owners that have currently seized capacity units of this resource. If the owner has not seized the resource then the value 0 is returned.

Search Step Enhancements

Search step enhancements have been added to allow the searching of new collection types including TransporterList, NodeInboundLinks, NodeOutboundLinks, NetworkLinks, ResourceOwners, and ObjectPopulation. Link-related searching may be useful for special node-oriented process logic as part of logical decision making.

The Search step also now provides both Starting Index and Ending Index properties to narrow the range of items to search within the collection. Previously, only a Starting Index option was provided. The Starting Index and Ending Index properties are also now available for the 'MinimizeReturnValue' and 'MaximizeReturnValue' objective-oriented search types.

'Owner Type' Enhancements on Steps

The Seize, Release, Move, Produce, Consume and Search steps now have an additional 'SpecificObject' option for the 'Owner Type' property that allows the owner that the step's action is in behalf of to be specified as a specific object reference. Before, the owner always had to be either the parent object containing the step or the associated object of the token executing the step.

This kind of feature provides more flexibility in process logic design. For example, an entity may go through a server, require a resource for processing and then the entity may wish to continue to another station without immediately releasing the resource. Process logic elsewhere in the model, regardless of whether the token executing the process is associated with the entity, can now easily command that entity to release the resource later in time.

Transfer and EndTransfer Steps 'Object Type'

The Transfer and the EndTransfer steps now provide an additional 'SpecificObject' option for the 'Object Type' property that allows the entity being transferred to be specified as a specific entity object reference. Before, the entity always had to be either the parent object containing the step or the associated object of the token executing the step.

This kind of feature provides more flexibility in process logic design.

Travel Step 'Token Wait Action'

The Travel step now includes a 'Token Wait Option' property that can be set to either 'None (Continue)' or 'WaitUntilTravelCompleted'. This kind of feature provides more flexibility in process logic design, by allowing additional control of a token's flow through process logic.

Properties Enhancement

When a property for a model is specified, under the Appearance section of properties, the Display Name and Category Name may be specified to group the various properties. You are also now able to specify a

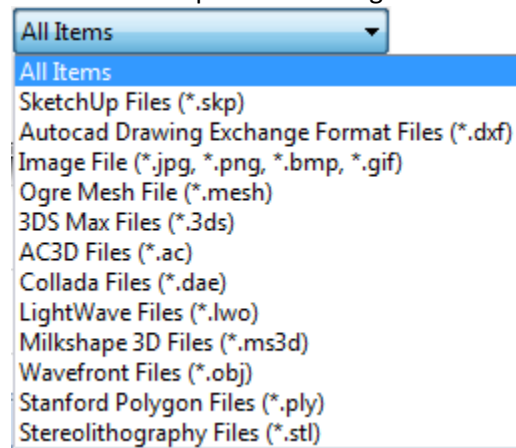
'Parent Property Name' such that one property is listed underneath another related property and accessible using the '+' on the parent property.

Shadows Change in Visibility Ribbon

The Shadows icon within the Visibility ribbon has now become two separate icons, including Direct Shadows and Diffuse Shadows. The **Direct Shadows** icon shows or hides shadows from a directional light source, like the sun. The **Diffuse Shadows** icon shows or hides soft shadows from other objects interacting with the illumination of an object. Only one of the shadows icons may be on at a time.

Expanded List of 3D Import Formats

We can now import from a larger number of 3D import formats, including the following:



Simio Enterprise Edition – Gantt Changes

Within the Simio Enterprise Edition, there have been several changes to make the entity and resource Gantt charts easier to use. When an object has 'Resource Logging' set as True, two additional properties are available. The Display Category property allows for hierarchical arrangement of resources within the Resource Plan Gantt (for example to display all resources of a given type or department grouped together). The Display Color property allows for user specified colors within the Resource Plan Gantt.

Additionally, if an additional column is added to the Resource Usage Log window for grouping (previous versions had this feature for Resource Gantt grouping), this column may be referenced within the Entity Workflow Gantt. This allows the Gantt chart for entities to be 'grouped' into various categories, as specified via the resource usage log (which typically may reference a data table column). See the Help page titled Resource Usage Log (Enterprise) for an example.

Flow Library Tank Object Enhancement

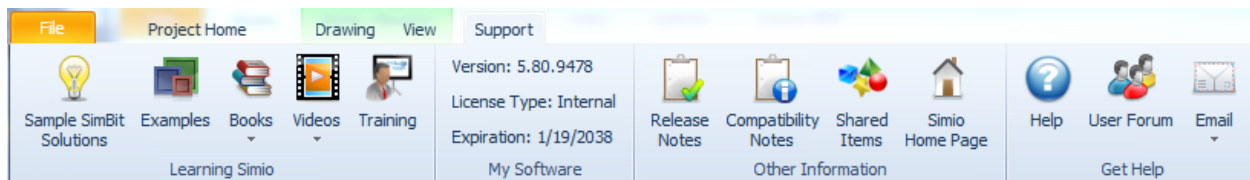
The Tank object within the Flow Library now has a 'State Assignments' section of properties to allow you to specify state assignments when the tank gets full or empty or its level rises or falls below a tank mark without having to go to add-on processes to do so.

Simio Release 5 – Sprint 80-81 – December 6, 2012

We are happy to announce the release of Simio Version 5. If you have been keeping up with interim (mostly internal) sprints, you will find that the most recent set of features is focused on many animation, GUI, and ease-of use features like the support ribbon. If you haven't seen Simio since the last major update (version 4) then you can look forward to reading the following 50 pages of release notes to answer the question "What's new?"

New Support Ribbon

A new ribbon has been added to the product that provides various support features that were previously found on the Start Page, in addition to numerous other resources for your use. This includes access to SimBits, Examples, and Books such as Introduction to Simio, Reference Guide and API Reference Guide, as well as links to videos, user forum and support email.



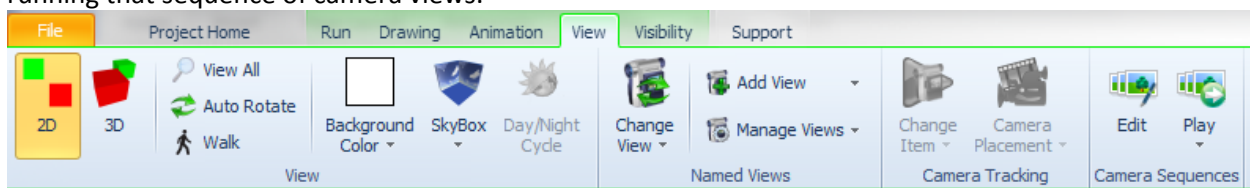
Additionally, information regarding customer license type and version number and maintenance expiration can be found on this ribbon. We are sure you will find this product information useful.

When your maintenance or your product license itself is close to expiring, we now also provide a reminder message on the bottom status bar to help you avoid inconvenient expirations.

Camera Sequences and HotKeys on View Ribbon

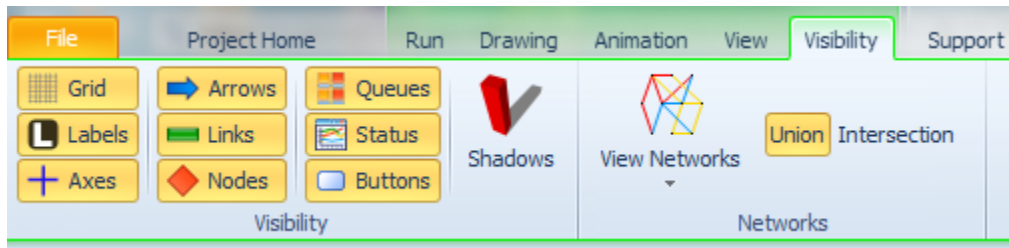
Two new buttons have been added to the View ribbon, including **Edit** and **Play** under **Camera Sequences**. A camera sequence is a sequence of instructions that can be played back either by selecting the sequence using the Play button or using a hotkey associated with the sequence. Named views, as well as the seconds to stay at that view and the amount of time to change views are specified for a given sequence. This sequence can then be used during the simulation run to demonstrate various views of the animation. The AirportTerminal example model contains an example of the new Camera Sequence. Simply click the Play button on the View Ribbon to see the camera sequences animated.

Both Named Views and the new Sequences support HotKeys. When adding/editing a named view or a camera sequence, you can now enter a HotKey as well as the Name. When a Named View has a HotKey assigned to it, hitting that key while in the Facility window, will take you to that Named View. When a Camera Sequence has a HotKey assigned to it, hitting that key while in the Facility window, will start running that sequence of camera views.



New Visibility Ribbon

With the addition of the above Camera Sequences buttons, we have moved some of the functionality from the View ribbon to a new Visibility ribbon. The ability to toggle on/off the visibility of the various items, such as grid, labels, arrows, nodes, etc., as well as network functions, have been moved to the new Visibility ribbon, as seen below.



F11 – Full Screen Viewing

The function key, F11, now allows the capability of collapsing all of the ribbons and other windows so that the simulation animation can be seen on the full screen. Pressing the F11 or Esc key while in full screen will then put back the ribbons and windows that were collapsed. This key can also be used in other windows, such as Processes, Data (for tables), Results, and Experiments.

Video Recording

Two new buttons have been added to the View ribbon to permit generation of video (.avi) files directly from within Simio. The Video group contains the Record Run Animation and Stop Recording buttons. Clicking on the Record Run Animation button will prompt the user to choose a name and location for their .avi file. The user will also be prompted to select a video compressor.

This capability can be used to generate simple videos of your animation files, more complex “how to” videos involving model-building, or even interaction with other applications. It is particularly useful when combined with the Camera Sequences option described above.

New Textures and Symbols

Six additional skins (textures) were added for more choices in representing rocks and minerals. You can also add your own skins to Simio simply by copying a jpg or png file into your Skins Simio folder (typically C:\Users\Public\Documents\Simio\Skins) and restarting Simio.

Many new symbols were added to the built-in library including new machines, various types of containers and storage places, and additional vehicles for hauling bulk materials and liquids.

Travel Step Enhancement

The Travel step now includes *Acceleration* and *Deceleration* properties for free space travel. This enables users to specify the rate at which an entity will gain speed going from a slower movement rate to a faster movement rate, as well as the rate at which the entity will lose speed going from a faster movement rate to a slower movement rate.

Transfer Step Enhancement

The Transfer step now has an additional property when transferring an entity into FreeSpace. This includes the *Facility Name*, which is the object whose facility free space into which the entity is to be transferred.

Lock Edits Enhancement

All items within the Facility window are now able to be right clicked and have 'Lock Edits' turned on/off. This will enable you to lock any object into place so that it does not move when navigating around the Facility window.

Ability to Duplicate Experiments

We have added the ability to duplicate entire experiments, which will duplicate the controls and data within the experiment but contain no results. Duplicating an experiment can be done in two ways. Through the Navigation window, the right click menu now allows you to duplicate, rename, or delete an experiment, as well as change experiment properties. Alternatively, in the Project window (by selecting the Project Name in the Navigation window and selecting the Experiment panel), right click now includes the ability to duplicate the experiment.

Option to Change Object References in a Model

When right clicking on a model within the Navigation window, the ability to change the model's object references is now available. This will open up a dialog displaying all the internal references the Model currently holds. From here, if a user placed two things down of the same name, they can change the name used to reference one in that particular model. They cannot change the reference names of objects in the same project, only things brought in externally (or objects that were in the project, but then they deleted).

New Libraries on the Simio User Forum

Like all Simio-provided libraries, these libraries are open so you can look at how they were built, learn from them, and subclass your own objects to improve them. But unlike our built-in libraries, these are not fully supported (although we can provide some help) and we may not continue enhancing them, particularly if we decide to build this capability into the base Simio products. We are anxious to receive your feedback on both libraries.



You can find these libraries via the **Shared Items** button on the Support Ribbon or in the Simio User Forum at <http://www.simio.com/forums/viewforum.php?f=36>

We encourage you to post your own contributions in this forum.

Extended Flow Library

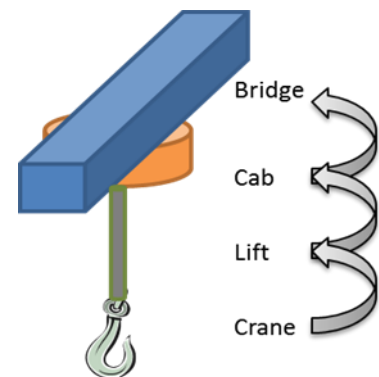
We have added an Extended Flow library and examples to the Simio User Forum. (<http://www.simio.com/forums/viewforum.php?f=36>). This library includes objects such as Solidifier, Liquefier, Filler, Extractor and Stockpile. A Solidifier can be used to convert a specified quantity of flow into discrete entities. A Liquefier converts a discrete entity into an outbound flow. The Filler is similar to a Combiner that combines input flow of a specified volume with a discrete entity and the combined entity departs the Filler. The Extractor is the flow equivalent of the Separator. It will separate a batch member entity from the parent entity, with the flow transfer of the batch member entity then discrete transfer of the parent entity once flow is complete. The Stockpile object is similar to a Tank with different animation graphics.

As we receive customer feedback on both the Flow Library and the Extended Flow Library we will probably build more of this type of capability directly into the Flow Library. But in the interim, hopefully this extended library will help in your modeling.

Crane Library

The Simio Crane Library is a collection of objects designed for modeling multiple cranes operating simultaneously in a bay. The library is provided as an example of complex material handling using the standard features of Simio. The Crane library may be used in conjunction with the Simio Standard Library, and Crane pickups are done using the standard TransferNode (e.g. the output side of a Server). Crane drop-offs can be done at either a BasicNode or TransferNode. The Crane Library can also be used with custom libraries as long as they support rider pickups using the standard transporter ride features.

The Crane Library consists of objects representing the **Bay**, **Bridge**, **Cab**, **Lift**, and **Crane** (the end effector that actually picks up and drops off the item). These objects are combined together to model multiple cranes moving in a single bay. The separation of objects in this way allows you to use your own symbol for each component. A Crane movement occurs by first rising up from the pickup node to a specified travel height, traveling laterally at that height, and then lowering down to the specified drop-off node. All travel is done through free space without the need to explicitly draw a network. The Crane library also fully supports independent acceleration/deceleration and the ability for one crane to cause another blocking crane to move out of the way.



The initial posting of the crane library has a few known issues and additional features already under development. We recommend deferring production use until these issues are addressed (January 2013), but we encourage early adopter use and feedback to help us identify features and other areas where we can improve the library.

Simio Release 4 – Sprint 79 – October 29, 2012

This sprint includes the official 'Beta' release of the Flow Library. We've made some additional enhancements to the objects and now also have a 6 model SimBit, FlowConcepts.spfx included in the software. Several user-requested enhancements have also been made to this sprint, including a method for clearing statistics, an expansion of table searching capability and an option for orienting entities in free space.

New ClearStatistics Step

A new ClearStatistics step has been added to the Processes window. This step allows users to clear all model statistics or specific object/element statistics. Possible uses for this step include wanting to clear all statistics during an interactive run based on some triggering event or 'warmup' time, or wanting to collect and report periodic statistics (e.g., some logic that every hour writes out some object or element statistics and then clears them).

Search Step Enhancement

The Search step now has a new property within the Advanced Options section that allows you to *Search Related Rows Only* (True/False). If the searched table has a primary key / foreign key relationship, this property indicates whether or not to include only related rows in the search. If this property is 'False', then the search will include all rows in the table, regardless of table relationships. The previous sprint behavior only searched those related rows.

Travel Step Enhancement for Free Space Movement

The Travel step for free space travel now includes the option for auto orienting the moving object towards its destination. This option is used to automatically orient the entity such that its front is facing the travel destination. Otherwise, the entity will stay in its current orientation when initializing the travel movement.

Stacked Bar Animation Enhancements

The Stacked Bar on the Animation ribbon has been enhanced to include 'Floor Display' options that will display tank contents on the floor so users can see them in 2D. A gauge like display can be shown on the left or right side of the tank, if desired.

Flow Related Changes

FlowNode Object and Regulator Element – Initial Output Entity Type and Function

The FlowNode from the Flow Library (as well as the related Regulator Element) now have a function for 'FlowNode.FlowRegulator.CurrentDesiredOutputEntityType' (or for the Regulator element, 'Regulator.CurrentDesiredOutputEntityType') that allows dynamic changing of the entity type when leaving the node (or regulator). The related property name has been changed to be the *Initial Output Entity Type*.

FlowNode Object – Dynamic Updating for Maximum Flow Rate and Yield Factor

The FlowNode now includes a Dynamic Updating section of properties. These properties include the *Maximum Flow Rate Equation* and the *Output Yield Factor Equation*. These two properties allow optional dynamic updating of either maximum flow rate and/or the output yield factor of the node's flow regulator. The new *Update Interval* property allows you to specify the time interval of the updates.

New Flow SimBit

We have added a new SimBit to our SimBit library:

- **FlowConcepts.spfx** – This SimBit includes six (6) unique models that show various concepts using the Flow Library. These include the following:
 - **SimpleFlow** – demonstrates flow from a FlowSource object, through a FlowConnector, to a FlowSink object.
 - **FillingEmptyingTank** – demonstrates filling and emptying a tank object.
 - **TransferringFromOneTankToAnother** – demonstrates transferring from one tank object to another.
 - **MergedFlow** – demonstrates merging two different entity flows.
 - **SplitFlow** – demonstrates splitting a single entity flow into two entity flows.
 - **CreateDiscreteEntitiesBasedOnFlow** – demonstrates the use of monitors and events to create discrete entities when a tank reaches a certain flow level.

Simio Release 4 – Sprint 78 – October 12, 2012

The long anticipated “official” Flow Library is now available! We’ve also enhanced the experiment window including adding drop-down lists for non-numeric fields, such as states, tables, and object references.

Flow Library

Unless you are using Express Edition, which is limited to using only the Standard Library, you will now see a second library option immediately beneath the Library objects. This is a beta version of Flow Library Version 1.0. We are still fine tuning it (e.g., object icons and symbols are still in process), but at this point it is stable and ready for use. We think you will find that this library establishes an excellent foundation & framework for fluid/mass flow based modeling. And as you can see in the brief description below, there is a great deal of power and flexibility built into those objects. We will be adding additional objects based on designs already in early planning stage as well as future customer requests. Please give us your Flow Library feedback to help meet your needs.

Initial Set of Flow Library Objects

FlowSource – The FlowSource object may be used to generate a flow of fluid or other mass of a specified entity type. The source of supply is by default considered to be infinite, but alternatively may be configured to run out if a stopping condition is met.

The flow rate out of the source is regulated by its 'Output' flow node.

FlowSink – The FlowSink object may be used to destroy the flow of entities representing quantities of fluids or other mass that have finished processing in the model.

The flow rate into the sink is regulated by its 'Input' flow node.

Tank – The tank object may be used to model a volume or weight capacity constrained location for holding entities representing quantities of fluids or other mass. Features included in this initial design include:

- The ability to specify either a volumetric or weight based capacity for the tank.
- The ability to specify initial contents in the tank of (possibly) varying entity types.
- An *Auto Refill Mode* that allows you to easily specify logic for automatically refilling the tank when it becomes empty, when its contents level has become low, or when some other specified event occurs.
- ‘Input’ and ‘Output’ flow nodes to easily connect mass or fluid flow connections into or out of the tank. The flow rates into and out of the tank are regulated by these nodes.
- The ability to specify ‘Low’, ‘Mid’, and/or ‘High’ tank level marks, and optional add-on process triggers to perform some action when the tank level rises above or falls below a mark.
- Optional add-on process triggers to perform some action when the tank becomes full or empty.

- A *FlowContainer.Contents.Volume* level variable that may be monitored or referenced to return the current volume level in the tank.
- A *FlowContainer.Contents.Weight* level variable that may be monitored or referenced to return the current weight level in the tank.
- A *FlowContainer.CurrentVolumeFlowIn/Out.Rate* and *FlowContainer.CurrentWeightFlowIn/Out.Rate* state variables that may be monitored or referenced to return the current total volumetric or weight based flow rates into or out of the tank.
- A default 2D & 3D level animation of the volume or weight contents in the tank.

FlowNode - The FlowNode object is a node specifically designed to regulate the flow of entities representing quantities of fluid or other mass. Common applications of a FlowNode object include use as an 'Input' or 'Output' node for controlling flow into or out of another object (such as the FlowSource, FlowSink, or Tank objects). A FlowNode may also be used to model a flow control point in a network of links. Features included in this initial design include:

- The ability to specify whether the maximum flow rate for the flow node's regulator is volumetric or weight based.
- The ability to specify the initial maximum flow rate for the flow node's regulator. You can then dynamically adjust that maximum flow rate for process control logic purposes by assigning the node's *FlowRegulator.CurrentMaximumFlowRate* state variable a new value.
- The ability to completely enable/disable a flow node's regulator, equivalent to turning a 'pump' at the node on or off. A disabled regulator will not output flow regardless of its current maximum flow rate setting (i.e., regardless of how open the 'valve' is). Disabling/enabling a flow node's regulator may be done by assigning the node's *FlowRegulator.Enabled* state variable a new value.
- The ability to specify an *Output Yield Factor* for the flow node's regulator. This factor is entered as a ratio of outflow to inflow, and may be used to scale the flow into the regulator such that there is a physical loss or gain represented in the regulator's output flow. You may specify an initial output yield factor as well as dynamically change the output yield by assigning the node's *FlowRegulator.CurrentOutputYieldFactor* state variable a new value.
- The ability to specify an *Output Flow Entity Type* for the outflow produced by the node's flow regulator. **Development Note:** This feature in Sprint 79 is already planned to be changed to *Initial Desired Output Entity Type*. And the flow node will then provide a *FlowRegulator.CurrentDesiredOutputEntityType* state variable that can be dynamically assigned a new value.
- Provides *FlowRegulator.CurrentVolumeFlowIn/Out.Rate* and *FlowRegulator.CurrentWeightFlowIn/Out.Rate* state variables that may be monitored or referenced to return the current total volumetric or weight based flow rates into or out of the node's flow regulator.
- Input Flow Control: Both 'Single Flow' and 'Merge Flow' control modes are provided.
 - The 'Single Flow' control mode allows you to process inflows only one inbound link at a time, and prioritize those inflows either by inbound link weight or by flow entity priority. There is also a *Switch Control Variable* option that easily allows you to define a discrete variable for use as a switch control mechanism to dynamically change the current

inbound link selection (e.g., to manually 'switch' from using one inbound link to using another inbound link).

- The 'Merge Flow' control mode allows you to merge flow by only matching inflows that have the same entity types or by merging all inflows regardless of entity type. There is also a *Merge Allocation Rule* that provides a variety of options for how competing inflows into the flow node should be merged. Choices currently include merging flow proportionally based on inflow rates, proportionally based on fixed link weighted percentages (blending), or merging using a preferred order strategy where some inbound links have higher priority for merging than others.
- Output Flow Control: Both 'Single Flow' and 'Split Flow' control modes are provided.
 - The 'Single Flow' control mode allows you to send outflow to outbound links one link at a time by either outbound link weight priority or by shortest path to an assigned destination. There is also a *Switch Control Variable* option that easily allows you to define a discrete variable for use as a switch control mechanism to dynamically change the current outbound link selection (e.g., to manually 'switch' from using one outbound link to using another outbound link).
 - The 'Split Flow' control mode allows you to split flow using a *Split Allocation Rule* with a variety of options. Currently only two options are provided to either split flow proportionally (fixed percentages) based on outbound link weights, or to split flow using a preferred order strategy based on outbound link weights (e.g., send flow first to a primary line but if any overflow then send that to a secondary line).

FlowConnector – The FlowConnector object may be used to define a direct, zero travel distance connection from one flow node location to another. A *Selection Weight* property is provided to define priority or percentage based information for the connector, and will be used if the connector is attached to flow node with an input or output flow control mode requiring such link weight information.

*General Note – The links within the Standard Object Library (such as Paths and Conveyors) can also be used somewhat with flow nodes, but if trying to use a Path be sure to set the "Allow Passing" property to "False". Otherwise, the model run speed will likely be very poor.

Varying Shape for Flow Links

We have added the elliptical and trapezoidal shape for all links, such as Paths, Conveyors, and Connectors. These shapes affect only the flow type entities and the logical flow calculations (for example, an elliptical shaped link of width 1 and height 1 will carry less than a rectangular one). For trapezoidal shapes, an additional property, Trapezoid Ratio, is available. This value is the ratio of the top of the trapezoid to its bottom (which always set to the width of the Link). It can be any non-infinite value ≥ 0 . A value of zero makes a triangle, a value of 1 makes a rectangle.

The type of shape will affect not only the flow calculations, but the shape of the available area graphic for the link, and the shape of the entities that move through the link.

Experiment Enhancements

In order to enable additional functionality, we have changed to a new technology underlying the experiment design window. You might notice some subtle changes.

- A new checkbox has been added in the heading of the first column. This permits easily toggling the active state on and off of all of the scenarios or any selected subset of scenarios. This is particularly useful when you want to only show a subset of the scenarios in the Response results window.
- Column sorting has been moved to a right click menu to allow column selection without changing sort order.
- The controls within the Experiment are based on the properties in a given model. Because these can be of various types (i.e., Tables, States, various Elements, Objects, etc.), these controls now display the drop-down lists available for each category from within the model. This will allow the user to much more easily select the control values for experiments.
- The responses within an experiment now include the Unit Type and Units for the response Expression. This allows users to display response results in varying display units. There are Units properties for the Upper Bound and Lower Bound of the response as well.

Simio Release 4 – Sprint 77 – September 18, 2012

We have added quite a number of new features to our Regulator element, in addition to adding statistics for Regulators, Containers and Links. These changes, along with continued development on our flow library objects, are moving us toward our release of the Flow Library at the end of the year. We've also made a significant enhancement to our experiments to allow for multi-objective optimization analysis – Simio is the first simulation software to feature this capability.

OptQuest Enhancements

When using the optional OptQuest for Simio add-in, the *Objective Type* property is now available to select either 'Single Objective', 'Multi-Objective Weighted' or 'Pattern Frontier'. This provides significant new flexibility in optimization.

Single Objective – OptQuest determines the set of control values that optimizes the experiment's Primary Response. This option is how OptQuest has worked in past sprints.

Multi-Objective Weighted – OptQuest optimizes across all responses with Objective set to Minimize or Maximize, taking into account each response's Weight value, to determine a single 'optimal' solution.

Pattern Frontier – OptQuest optimizes across all responses with Objective set to Minimize or Maximize, and finds the set of scenarios that are optimal, rather than a single 'optimal' solution based on weights.

* Note that the HospitalEmergencyDepartment example has been updated to incorporate the new multi-objective analysis using OptQuest.

New Entity Function

CurrentConnectorLink - If the entity object is currently using one or more connector links to transfer between two node locations, then this function returns a reference to the forward most connector being used. Otherwise the Nothing keyword is returned.

Note that a connector link does not have a logical length and thus is not considered to be physically occupied by an entity that is using it. To get a reference to the link that an entity's leading edge is currently physically located on, use the 'CurrentLink' function.

Flow Enhancements

We have continued with our flow enhancements including the following:

Flow Statistics

Container – Flow In and Flow Out of a container are now displayed in the simulation results.

Regulator – Flow Out is reported if it is greater than 0, while Flow In is reported if it is not equal to Flow Out (yield factor).

Link – Flow In and Flow Out are reported if at least one of them is greater than zero.

Regulator Enhancements

Initial Output Yield Factor – This new property should be entered as a ratio of outflow to inflow and may be used to scale the flow into the regulator such that there is a physical loss or gain represented in the regulator's output flow.

Output Flow Entity Type – This property now applies to single or merged flow from the regulator and is used to specify the entity type of the outflow produced by the regulator. If unspecified, then the output flow entity type will be determined by the inflow entity types.

Merge Allocation Rule – The *ProportionalBasedOnExpression* rule has been added, along with an additional related property *Merge Proportion Expression* to return the desired merge proportion for a flow request in the regulator's flow request queue.

Simio Release 4 – Sprint 76 – August 29, 2012

In Sprint 76, we have made many changes to various parts of the software including functions for additional flexibility in network travel, enhancements to the stacked bar, regulator and container for flow processing and additions to the Simio Enterprise edition.

New Link Functions

StartingNode = Returns a reference to the node located at the drawn starting point of this link.

EndingNode = Returns a reference to the node located at the drawn ending point of this link.

Network Element Functions

Distance(fromNode,toNode) Returns the shortest path distance between two nodes using this network.

NextLink(fromNode, toNode) Returns a reference to the next link on the shortest path between two nodes using this network.

NextNode(fromNode, toNode) Returns a reference to the next node on the shortest path between two nodes using this network. If the destination node is returned, then the two nodes are directly connected by a link.

PathExists(fromNode, toNode) Returns 'True' if a followable travel path exists between two nodes using this network.

New Entity Function

LastLinkUsed - Returns the most recent link used by the entity to complete a travel or instant transfer (in the case of a connector link) into a new node location. The value returned by this function is updated whenever the entity transfers from the end of a link into a node, or transfers across a connector from one node into another node.

Animated Stacked Bar Enhancements

The Stacked Bar graphic on the Animation toolbar has been enhanced to allow various 'types' including Container, Queue and Expressions (prior version animated only Queue contents).

- **Container:** When displaying a stacked bar for a Container, only the container element name is required and the volume contents of the container are displayed. The total volume capacity of the container then defines the outside portion (Total Expression) of the stacked bar.
- **Queue:** When displaying the Queue, a Total Expression property has been added to describe the total space available in the graphic.
- **Expression:** When displaying a stacked bar for Expressions, a repeat group of Expression and Color are available to graphically show various expression values within the stacked bar. This could include various model states or expressions. A Total Expression can be used to describe the total space available in the graphic.

Flow Enhancements

We have continued with our flow enhancements including adding infinite flow rates on the Regulator element, outbound link rules, merging within Containers and merging/transfers to different destinations with the Regulator. These changes will allow us to continue developing our 'flow library' which is intended to be released at the end of the year.

Simio Enterprise Edition Enhancements

Within Simio Enterprise Edition, several changes have been made to allow for comparison of various model alternatives. After a user performs the 'Analyze Risk' within the Operational Planning ribbon, the Results panel option on the left side allows for viewing of various results, including Target Summary, Target Detail, Risk Plots and others. Within the Target Detail tab, there are options on the Operational Planning ribbon for Plan Comparisons. These include 'Save for Comparison' and 'Show Differences' buttons.

The '**Save for Comparison**' button will save the current model (including logic, data tables, etc.) and place the saved information in a 'Baselines' folder in the Navigation window, named by the ModelName_Date_Time.

Once at least one baseline has been saved, the '**Show Differences**' button becomes available and acts as a toggle between the Target Detail report for the baseline and a Target Detail report that shows the 'Differences' or 'Changes from Baseline' between the two models. This allows users to compare various models and discover what difference, if any, the model changes make to the final target results.

Simio Release 4 – Sprint 74-75 – August 9, 2012

A major new feature in Sprint 75 is the addition of many enhancements for free space movement. This allows much more flexible free space travel for entities, vehicles and workers. We have also continued to work on Flow Processing enhancements, including the addition of flow states for Containers, Regulators and Links.

Free Space Movement Enhancements

Often you may want to travel between many locations, but don't want to manually interconnect every possible node combination. While using a network with shared links saves a lot of time, it can sometimes still be tedious. In the case where you are looking for straight-line movement between nodes, Simio now has a new solution: **Free Space Movements**. While Simio has always supported movement through free space, it has now been made much easier with the addition of the Travel Step, updates to the SetNode and SetNetwork steps, enhancements to the Entity object and the Node, Worker and Vehicle library objects, and a host of new functions. In addition this feature provides support for easier detailed free space movements such as those you may want to include in a custom object or library.

New Travel Step

A new Travel step has been added for use in the Processes window. This step may be used to do a direct, straight-line movement of an entity in free space to a specified location. The executing token will be held at the Travel step until the travel movement is completed. The new destination may be specified as a fixed object, absolute coordinates or relative coordinates to the entity's current location. A maximum desired movement rate and ending movement rate may also be specified.

Entity Objects (including ModelEntity, Worker, Vehicle)

The *Initial Network* property now provides the choice 'No Network (Free Space)' so that free space movement of entities, workers and vehicles may be incorporated. The default value of this property still remains 'Global'.

There are now new OnEnteredFreeSpace standard processes for the Worker, Vehicle and ModelEntity objects. Please see the Known Anomalies regarding information on free space travel for entities.

New Node Functions

- **IsInputNode** – Returns 'True' if this node is an external input node for entering another object. This function replaces the obsolete function, IsExternalInputNode.
- **IsOutputNode** – Returns 'True' if this node is an external output node for exiting another object.
- **Nearest.Node** – Returns the node that is the shortest straight-line distance from this node.
- **Nearest.InputNode** – Returns the input node that is the shortest straight-line distance from this node.

- **Nearest.OutputNode** – Returns the output node that is the shortest straight-line distance from this node.

Step Changes

- **SetNetwork step** - *New Network Name* property now provides the choice ‘No Network (Free Space)’. This lets you dynamically change an entity or transporter from using a network to only traveling in free space and vice-versa.
- **SetNode step** – If setting a specific node name, the *Node Name* property now provides the choice ‘No Destination’ (similar to <null> in prior sprints).

Free Space SimBits Changes

We have updated our SimBits library with changes based on the addition of the free space travel enhancements listed above. Two SimBits, *EntityMovementInFreeSpace.spfx* and *MovementInFreeSpaceWithATargetLocation.spfx* have been deleted and replaced by:

- **FreeSpaceMovement.spfx** – This SimBit displays the use of various types of free space travel. There are three models within the project. The first demonstrates entity free space movement from a Source to multiple Servers then to a Sink. The second demonstrates entity free space movement between two nodes after using network based travel (including the new Travel step). The third model demonstrates both entity and vehicle free space movement.

Flow Enhancements

Simio has always provided support for discrete and continuous flow, but a set of changes to the Simio engine (e.g., the Steps, Elements, and other related low-level changes) are nearly complete that will set a new standard of modeling capability. Examples of using these new features illustrated with a sample library can be found on the Simio Insiders portion of the user forum, under the SI Material Handling topic: <http://www.simio.com/forums/viewtopic.php?f=10&t=762>. While these low-level features are nearly complete, we are anxious to hear your feedback at <http://www.simio.com/forums/viewtopic.php?f=10&t=710> as we continue with our design and implementation of the “official” flow library that is expected to be completed late this year.

New Link States

- **CurrentVolumeFlowIn** – State used to get the current total volume flowed into a link.
- **CurrentVolumeFlowOut** – State used to get the current total volume flowed out of a link.
- **CurrentWeightFlowIn** – State used to get the current total weight flowed into a link.
- **CurrentWeightFlowOut** – State used to get the current total weight flowed out of a link.

New Regulator States

- **CurrentMaximumFlowRate** – State used to get or set the current maximum flow rate of a regulator.
- **CurrentVolumeFlowOut** – State used to get the current total volume flowed out of a regulator.
- **CurrentWeightFlowOut** – State used to get the current total weight flowed out of a regulator.

New Container States

- **CurrentVolumeCapacity** – State used to get or get the current volume capacity of a container.
- **CurrentWeightCapacity** – State used to get or set the current weight capacity of a container.
- **CurrentVolumeFlowIn** – State used to get the current total volume flowed into a container.
- **CurrentVolumeFlowOut** – State used to get the current total volume flowed out of a container.
- **CurrentWeightFlowIn** – State used to get the current total weight flowed into a container.
- **CurrentWeightFlowOut** – State used to get the current total weight flowed out of a container.

Splitting Flow

You can now build a model with a node's Output Flow Strategy set to Split Flow and using an appropriate rule to allocate the flow between outbound links.

Miscellaneous Changes

- Infinite Regulator flow rates have been added
- Connector links for flow (links with zero length) are now supported

Other Miscellaneous Changes

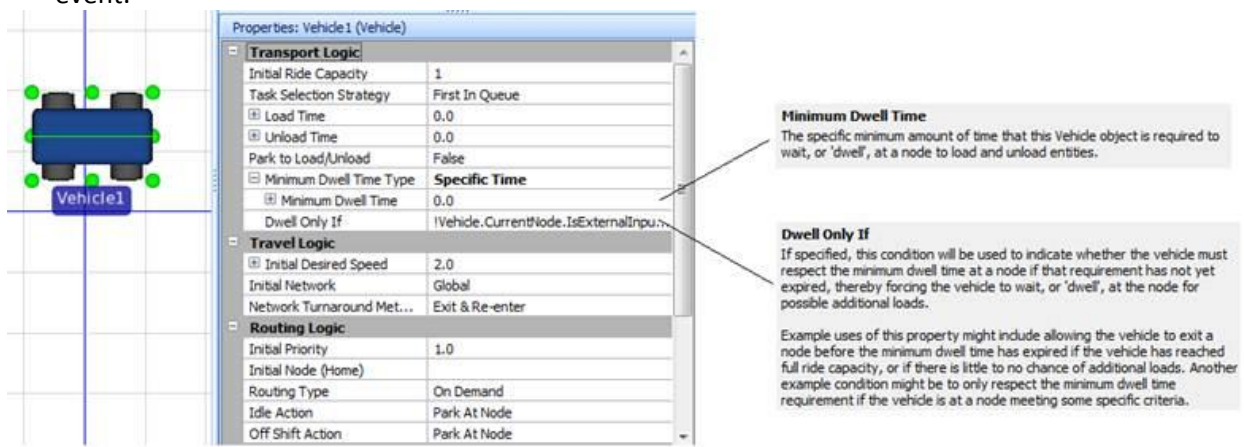
- Ctrl-Wheel now zooms within an individual process in the Processes window
- **AddRow step** – Now supports a 'SpecificObjectorElement' Object Type value, asking for the specific 'ObjectOrElement' if that option is specified.
- **Search step** – Improved trace when running the simulation model to include the exact items found by the search and the *Save Index Found* and *Save Number Found* values, if specified.

Simio Release 4 – Sprint 73 – June 26, 2012

The major new feature in Sprint 73 is an enhancement to Vehicle and Worker to easily support flexible Dwell Time. We also added new macros to support object development by more advanced users. We also made some subtle changes that improve overall product ease of use and robustness.

Vehicle & Worker

- **Minimum Dwell Time** – New capability to make it easier to control how long a vehicle will wait at a node before proceeding. The current default continues to be to wait only until all entities that can be immediately loaded are loaded. Now it is easy to add flexible waiting (“dwell time”) options to do such things as wait until full, wait until a scheduled departure time, or wait for a specific condition or event.



Minimum Dwell Time
The specific minimum amount of time that this Vehicle object is required to wait, or 'dwell', at a node to load and unload entities.

Dwell Only If
If specified, this condition will be used to indicate whether the vehicle must respect the minimum dwell time at a node if that requirement has not yet expired, thereby forcing the vehicle to wait, or 'dwell', at the node for possible additional loads.

Example uses of this property might include allowing the vehicle to exit a node before the minimum dwell time has expired if the vehicle has reached full ride capacity, or if there is little to no chance of additional loads. Another example condition might be to only respect the minimum dwell time requirement if the vehicle is at a node meeting some specific criteria.

Property Default Values

Macros – There are 'macros' that can be put in the default value of a property that will expand to the correct value when an object when that property is instantiated. The new macros are:

- \$(InstanceName)
- \$(DefinitionName)
- \$(AssociatedObjectInstanceName)
- \$(AssociatedObjectDefinitionName)

For example previously if you renamed ModelEntity to something else, then placed an instance, you would get an error about ModelEntity.Picture not being a valid property value. The default value for the property is now set to "\$ (DefinitionName).Picture", which now resolves correctly when the instance is placed, and no errors occur. This feature will make it easier to design library object which can be reused without errors like this.

Miscellaneous

Zoom – Zoom now works better in that zoom in, then out will return you to the original zoom level. And a problem with some “fuzzy” graphics on view change has been corrected.

Process Window – Categories now have a better look and feel as well as some small bug fixes.

Simio Release 4 – Sprint 72 – June 1, 2012

In this sprint, we have added many new entity functions in support of sequence planning and scheduling. These functions can be used to access various pieces of information within an entity's sequence table/plan. Additionally, we are continuing our development on flow processing. A new example library is now available on our user forum for a preliminary look at the various features within flow.

New Entity Property

- **Due Date Expression** – The expression used to return a 'due date' value for an entity of this type.

New Entity Sequence Related Functions

- **Entity.Sequence.NumberJobStepsRemaining** – Returns the current number of steps remaining in the entity's assigned sequence, including the current job step.
- **Entity.Sequence.ExpectedOperationTimeRemaining** – Returns the sum of the expected operation times for the remaining steps in the entity's assigned sequence, including the current job step.
- **Entity.Sequence.CriticalRatio** – Returns an index for determining how much the entity's assigned sequence is considered to be on schedule given the due date and expected operation time remaining. A value of 1.0 is 'on schedule'. A value less than 1.0 is behind, while a value larger than 1.0 is ahead of schedule. This function returns a value equivalent to the expression $(\text{Entity.Sequence.DueDate} - \text{Run.TimeNow}) / \text{Entity.Sequence.ExpectedOperationTimeRemaining}$.
- **Entity.Sequence.SlackTime** – Returns the expected amount of time that would be left until due date after completing the entity's assigned sequence, if the sequence's remaining operation time was started now. This function returns a value equivalent to the expression $(\text{Entity.Sequence.DueDate} - \text{Run.TimeNow}) - \text{Entity.Sequence.ExpectedOperationTimeRemaining}$.
- **Entity.Sequence.SlackTimePerOperation** – Returns the average slack time per remaining operation for the entity's assigned sequence. This function returns a value equivalent to the expression $\text{Entity.Sequence.SlackTime} / \text{Entity.Sequence.NumberJobStepsRemaining}$.
- **Entity.Sequence.DueDate** – Returns a due date for the entity to complete its assigned sequence. The 'Due Date Expression' property for the entity type is used to get the due date value.
- **Entity.Sequence.ModifiedDueDate** – Returns a modified due date value for the entity's assigned sequence, which is the highest of either the due date or the expected completion date if the sequence's remaining operation time was started now. This function returns a value equivalent to the expression $\text{Math.Max}(\text{Entity.Sequence.DueDate}, \text{Run.TimeNow} + \text{Entity.Sequence.ExpectedOperationTimeRemaining})$.
- **Entity.Sequence.DestinationNodes** – The destination nodes in the entity's assigned sequence.
- **Entity.Sequence.DestinationNodes.NumberItems** – Returns the number of destination nodes in the entity's assigned sequence.
- **Entity.Sequence.DestinationNodes.FirstItem** – Returns a reference to the first destination node in the entity's assigned sequence.
- **Entity.Sequence.DestinationNodes.LastItem** – Returns a reference to the last destination node in the entity's assigned sequence.
- **Entity.Sequence.DestinationNodes.ItemAtIndex(index)** – Returns a reference to the destination node at a specified step index position (or job step) in the entity's assigned sequence.

- **Entity.Sequence.DestinationNodes.IndexOfItem(node)** - Returns the one-based step index (or job step) of the first occurrence of a specified destination node in the entity's assigned sequence. If the entity does not have an assigned sequence then the value 0 is returned.

New Node Property

- **Sequence Expected Operation Time** – The expression used to estimate an expected operation time if this node is in the destination sequence of an entity's assigned sequence table.

Simio Release 4 – Sprint 71 – May 20, 2012

As in the past several sprints (which have not been released externally), we are continuing to work on features to support flow in future sprints. The Container and Regulator elements, as well as changes to the Transfer step, allow basic flow type objects to be designed and tested. We've also continued to make enhancements to our standard product, such as adding transporter functions and enhancing the Wait step.

New Transporter Functions

- **RideStationLoad** – Returns the current 'load' on the station location that holds the entities riding on the transporter. The ride station 'load' is defined as the sum of current entities reserving space on and intending to ride the transporter plus the current number of riders.
- **RideStationOverload** – Returns the current difference between the load and capacity values (a positive difference indicating an 'overload') for the station location that holds entities riding on the transporter. The ride station 'load' is defined as the sum of the current entities reserving a spot on and intending to ride the transporter plus the current number of riders. This function returns the difference between the load and the current ride capacity (overload = load – capacity).

Revised Entity Function

- **Entity.ReservedTransporter** – This function is now set to the reserved transporter until the entity is actually loaded and has moved into the transporter's RideStation. (Previously, this value was set to 'Nothing' during any loading time for the entity to move onto the vehicle). Thus, if an entity has reserved space on and intending to ride a specific transporter (but has not yet loaded onto it), its ReservedTransporter function is set to that transporter. The Entity.CurrentTransporter function is used once the entity is riding on the transporter.

Standard Library Changes

- **Vehicle/Worker** – The Ride Capacity property is now an expression property instead of an integer property. This allows greater flexibility in specifying the capacity for vehicles and workers. This property was also renamed *Initial Ride Capacity*.
- **TransferNode** – The transporter 'Selection Expression' default value is now 'Candidate.Transporter.RideStationOverload' instead of 'Candidate.Transporter.RideCapacityRemaining'.

Routing Group Element / Transfer Node Enhancement

The Routing Group element now supports the Node List property being independently evaluated for each entity route request (per table entry, for example). Within the Transfer Node, for example, if selecting the entity destination using the 'Select From List' option, the name of the node list might come from table data and differ based on the entity.

Ride Step Changes

The transporter 'Selection Expression' default value is now 'Candidate.Transporter.RideStationOverload' instead of 'Candidate.Transporter.RideCapacityRemaining'

Wait Step Enhancement

The Wait step now allows multiple events and event conditions to be evaluated before releasing entities from the step. The user has the ability to specify for the tokens to wait until all events have been fired or wait until any one of the events have been fired before releasing the tokens from the Wait step.

New SimBit

We have added a new SimBit to our SimBit library:

- **HourlyStatistic.spfx** – This SimBit displays the use of the Timer and TallyStatistic elements in calculating the average number of entities in a given input buffer on an hourly basis.

Flow Enhancements

The Container and Regulator elements have been added and the Transfer step has been revised to support the addition of Flow type objects in the future. The Container element is used to define a volume or weight capacity constrained location for holding entities representing quantities of fluids or other mass. The Regulator element may be used to regulate flow transfers of entities into or out of a location. The Transfer step has been modified to include transfers to and from Containers.

It is important to note that while we currently have some new elements and revised steps to support flow, these features should not yet be used within any production models. Examples of using these new features within an object and/or small library are available upon request. We strongly encourage your feedback through the Simio Insiders portion of the forum, under the SI Material Handling / Updated Flow Design, <http://www.simio.com/forums/viewtopic.php?f=10&t=710>.

Simio Release 4 – Sprint 70 – April 20, 2012

Again in this sprint, much of our development time has been spent with flow related features. We've enhanced a few of our node related functions, as well as added capability to our Search window. Some changes have been made to the way negative costs are handled within our costing areas as well.

New Node Functions

- **AssociatedNode** (description modified) – For an external node, this function returns a reference to the object that the node is attached to and which may be entered or exited using the node. If the node is not an external node, then the Nothing keyword is returned.
- **AssociatedStation**– For an external input node, this function returns a reference to the immediate station location inside the node's associated object that may be entered using the node. If there is no station location, then the Nothing keyword is returned.
- **AssociatedStationLoad**** – For an external input node, this function returns the current 'load' on the station locations inside the node's associated object that may be entered using the node. The associated station 'load' is defined as the sum of current entities en route to the node intending to enter the stations, plus the current entities already arrived to the node but still waiting to enter the stations, plus the current entities occupying the stations.
- **AssociatedStationOverLoad**** – For an external input node, this function returns the current difference between the load and capacity values (a positive difference indicating an 'overload') for the station locations inside the node's associated object that may be entered using the node.
 $\text{Overload} = \text{Load} - \text{Capacity}$.
- **NumberTravelers.RoutingInToEnterAssociatedObject** – For an external input node, this function returns the current number of travelers with their destination set to this node and which can enter the node's associated object.

**Unlike the deprecated InputLocation.Load and InputLocation.Overload functions, the new AssociatedStationLoad and AssociatedStationOverload functions will consider the 'Processing' station contents of standard library objects such as Server, Combiner, Workstation, and Separator.

Costing Changes

The object and cost center element Cost state can now have a negative value or rate. Previously, Simio was turning negative values into 0.0.

Additionally, we have several built-in cost drivers including resource usage rate/idle rate/per use costs, station per use/holding rate cost, for making it easier to accrue costs to cost centers and object costs. Previously Simio was turning negative values into 0.0 for all the built-in cost drivers. We have added run-time error messages on all of our built-in cost parameters if the user enters a non-negative value.

Search Window Enhancements

We have enhanced the Search window (Project Home ribbon) to include object definition names in the searches. For example, if you have 10 Server type objects in your model, but none of the with the word 'server' in the name, previously, a search would not have found these. Now, the Item Type is shown as 'Server' and all server definitions within a Facility window will be located.

Simio Release 4 – Sprint 69 – April 5, 2012

In this sprint, much of our development time has been spent with flow related features. We've also added some user requested features, such as allowing property values within schedules and applying the external view symbol to a changed symbol of an object.

Animation Ribbon – Stacked Bar Graphic

We have added a new graphic for displaying information about queue contents called a Stacked Bar. The Stacked Bar graphically displays the value of some expression evaluated for each item in a designated queue as a display of stacked blocks (for example, ModelEntity.Volume for a Server1.Processing.Contents queue). This can be used in the Facility window or External View window of an object.

New SimBits

We have added two new SimBits to our SimBit library:

- **AppointmentArrivals.spfx** –This SimBit shows model arrivals that occur at a specific time, such as a scheduled appointment. In the example, entities arrive every 15 minutes between 8:00 – 4:30, but also have a time deviation to demonstrate early and late arrivals.
- **WorkersArriveLateToShift.spfx** –This SimBit demonstrates the use of schedules for a worker, but also considers the fact that sometimes the worker arrives late to his/her shift. This is done by using a 'sub-classed' Worker object and adding additional properties for the worker's probability of lateness and late time.

New Container and Regulator Elements

We have added two new elements to the Elements panel that will be used in flow processing, including a Container and a Regulator. These elements are still under development.

External View Symbol

We added an option in the Right-Click menu when selecting an object that will allow the user to revert back to the 'external view' symbol originally specified. For example, perhaps you place a Server and change its symbol to another graphic. Later, you may wish to change the symbol back to its original default symbol, as shown in the External View of the object.

Schedule 'Value' and 'Cost Multiplier' Enhancements

Within a Schedule, under the Day Pattern section, the Work Periods specified allow the specification of a 'Value' which is the capacity value for a given resource. Additionally, a 'Cost Multiplier' can be specified that will factor the cost associated with that given work period and resource. Both of these fields may now be specified as an expression, which may include a property value. This will allow both fields to be changed through experimentation as well.

It is important to note that 'Value' and 'Cost Multiplier' fields are also available within various Exceptions specified (in the Work Period Exceptions within the Schedule, as well as within the Work Period Exceptions specifically within the object's properties). These currently do NOT support properties at this time, but should in the next sprint.

New Node Function

- **AssociatedObject** – For an external node, returns a reference to the associated object that may be entered or exited using the node. (For example, for Input@Server1 basic node, this function would return Server1 as its associated object).

Planning Tab – New 'Logs' View (Enterprise Edition)

Within the Planning Tab of the Enterprise Edition, you will see that the views have changed slightly to include a new 'Logs' view. We have taken the various logs, including the Resource Usage Log, Resource State Log and Constraint Log, out of the Entity Workflow and Resource Gantt areas and into their own view. This will enable users to move more easily between the various log files, without having to move from Entity Workflow to Resource Gantt and back, as was the case with the previous layout.

Simio Release 4 – Sprint 68 – March 15, 2012

In this sprint, we have continued to add features to Simio to support Flow Processing in the near future. These changes include the addition of weight and volume states (and rates) to entities, as well as some node internal changes to prepare for flow support. Many of our other changes are customer related enhancements, including the addition of arrays for string, element reference and object reference states, as well as variable expression threshold monitoring. We are continuously adding flexibility to our Simio Enterprise edition, including new features for Gantt sorting and filtering.

Entity State Changes

Entities now have Volume and Weight states. The volume is referenced as ModelEntity.Volume and represents the current *logical* volume of the entity. If not defined, the entity's volume will be the *implicit* volume as calculated by the Size parameters. The weight is referenced as ModelEntity.Weight and has an initial value as calculated by the Density and Volume of the entity. Volume and Density can be specified within the Entity's property grid. Any changes to weight (or volume) will change the volume (or weight) to keep a constant density of the entity. The ratio of the Size implicit volume to the logical volume is kept constant during the run.

A 'Rate' parameter is also available on the Volume and Weight states, as well as on the Size state under each of its Length, Width, and Height parameters. The Movement state of the entity also has this similar 'Rate' parameter. These 'Rate' states can be monitored as continuous variables.

Based on the above additions of volume and weight with the entity, the Station element's Contents queue now has the following read-only parameters as well – Volume, Volume.Rate, Weight, Weight.Rate. These represent the cumulative volume and weight contents of the queue. These can be used in a Monitor, so that the total volume or weight of objects in a station doesn't exceed a given value, for example.

String and Element Reference State Arrays

The String, Element Reference and Object Reference states can now be defined as arrays. In the past, these states were scalar values.

New 'Before Processing' Add On Process Trigger

Within the Server, Separator, and Combiner, a new 'Before Processing' add-on process trigger has been provided. This trigger occurs when the entity has been allocated the server/combiner/separator capacity, but before entering (or ending transfer to) the object's processing station.

Path, Conveyor and TimePath capacity Changes

The Path, Conveyor and TimePath objects in the standard library have been modified to have an Initial Traveler Capacity that can be an expression (previous software was 'Traveler Capacity' and was a static integer). This allows the dynamic changing of the capacity by using the 'NodeName.CurrentTravelerCapacity' state.

Node Changes

A node object in Simio now provides the following property:

- **Initial Traveler Capacity** – The initial maximum number of travelers that may simultaneously occupy the node.
- **Entry Ranking Rule** – The rule used to rank entry into this node among competing travelers.
- **Entry Ranking Expression** – The expression used with a ‘Smallest Value First’ or ‘Largest Value First’ entry ranking rule.

Nodes also have the following states:

- **EntryQueue** – The queue of entities that are waiting to enter the node.
- **CurrentTravelerCapacity** – State used to get or set the current maximum number of travelers that may simultaneously occupy the node.

Nodes also have the following functions:

- **NumberTravelers** – Returns the current number of travelers occupying this node.
- **NumberTravelers.RoutingIn** – Returns the current number of travelers with their destination set to this node.

The standard library node objects, `BasicNode` and `TransferNode`, are no longer ‘resources’; therefore, they no longer have the `Capacity`, `Ranking Rule` and `Ranking Expression` properties. The new properties listed above have replaced them. We’re confident that the priority merging flow through a node is working better with this new approach.

Note about behavior change within the `BasicNode/TransferNode`: Though the traveler capacity functionality now provided by `BasicNode` & `TransferNode` is very similar to the resource-based approach, one difference is that now an entity cannot actually enter the node and execute the ‘`OnEntered`’ process and the ‘`Entered`’ add-on process trigger until it actually is allocated traveler capacity and thus selected from the node’s `EntryQueue`. Before, an entity would always first enter the node, execute the ‘`Entered`’ add-on process trigger if specified, and then have to seize the node resource to continue ‘crossing’ through the node.

Auto-Completer and Drop-Down List Changes

We’ve taken non-resources off of ‘`Seize`’ drop-down lists and no longer show resource functions and states for non-resource objects in the auto-completer. This should make model building using lists easier to use and select from.

Monitor Element Enhancement

For Monitor elements that have a *Monitor Type* set to ‘`CrossingStateChange`’, the *Threshold Value* has now changed to *Initial Threshold Value*. The monitor now has a state ‘`MonitorName.CurrentThresholdValue`’ that can be changed during the simulation run. Previously, this threshold was a constant real value. This provides additional flexibility for users to have changing thresholds for a given Monitor element.

Gantt Chart Enhancements (Enterprise Edition)

If you change row heights or expanded states within the Gantts, Simio now tracks this and remembers it across re-runs.

Gantt charts can now be filtered by resource or entity by right clicking on the resource or entity and selecting the Filter to option. This will filter both the Gantt as well as the associated Log. This is available within both the Resource Plan Gantt, as well as the Entity Workflow Gantt.

Resource Usage Log Columns (Enterprise Edition)

The Resource Usage Log now has an option to add columns within the Log. These additional columns have the option of being displayed in either the Resource Gantt or the Entity Gantt. The column values are calculated based on an expression value. This is useful for showing additional related info in the Gantt charts.

For example, if you would like to sort the Resource Gantt in a particular manner, you can add a column to the Resource Usage Log that would return a sorting value (1, 2, 3, 4, etc.). This may be based on a table entry or any expression specified. The most recent values in the log for that resource are then shown in the Gantt and can be sorted on.

Simio Release 4 – Sprint 67 – February 17, 2012

In this sprint, we've continued to enhance the output reports and charts within Simio Enterprise Edition. Additionally, object referencing flexibility has been increased by providing more capability of how various functions are referenced. We've added multiple unit measures in anticipation of additional Simio features in future sprints.

Dynamic Objects within Resource Plan – Gantt (Enterprise Edition)

The Gantt chart within the Resource Plan panel of the Planning tab for Simio Enterprise has been enhanced to now include dynamic objects, such as Workers and Vehicles. The Gantt chart for these objects shows the order of entity processing, any constrained entities and resource state by individual unit within the population (i.e., Worker[1] and Worker[2] if the object named Worker has an *Initial Number in System* of '2').

States and Targets Displayed in Properties Window (Enterprise Edition)

The States and Target data that is generated for a given plan is now displayed within the Properties window for a given Entity using a particular Resource. This provides easier viewing of information about states and targets within a table in a grid format.

Object Referencing

We have added the ability to get a reference to an object's external input or external output node, which will allow the user to get property, state or function information on an object's external nodes if they have a reference to the object itself. For example, if you have a Server and you want to get a reference to its external input node, you could type 'Server1.Input.NumberRoutingIn' to determine the number en route to that input node. Alternatively, if you have an ObjectReferenceState that is holding a reference to Workstation1, then you could reference the queue for the output node by specifying 'ObjectReferenceState1.Workstation.Output.RidePickupQueue.NumberWaiting'.

The keyword 'Is' can now be used with an ObjectInstanceProperty so the user can decide if an object is the same type as the object specified in an object's ObjectInstanceProperty. As an example, if you have an object, let's call it 'submodel', and you put an ObjectInstanceProperty on that object, then you can now say Is.ObjectReferenceProperty in your logic. So, you might have an instance of Submodel in the Model and the ObjectInstanceProperty is set to 'PartA' but it has PartA and PartB going into it. The logic might contain a Decide Step that asks 'Is.ObjectInstanceProperty', which is essentially asking 'Is.PartA' but only because that instance's ObjectInstanceProperty is set to 'PartA'.

Units Settings

We've added a Units Settings button with pull down to set the units to display times, lengths, volumes, mass, travel rates, volume flow and mass flow. This button can be on many ribbons and affects the units displayed within the reports, pivot grid, watch window, status bar and trace window.

Simio Release 4 – Sprint 66 – January 30, 2012

In this sprint, we've added more flexibility within our Standard Library objects, as well as made some enhancements for our new Enterprise edition software. Work schedules have been enhanced to include a cost multiplier (for overtime, as an example), and have also been added to the Vehicle object. User requested enhancements include the addition of new functions for entities and a new EndRun step, which allows the user to dynamically end the simulation run. We've started adding Enterprise edition specific SimBits to the software as well.

Work Schedules for Vehicles

Vehicles have been enhanced to include Work Schedules, similar to Workers and other Standard Library objects. The Vehicle now includes an 'Off Shift Action' for determining its behavior when it goes offshift. Additionally, two add-on process triggers, Off Shift and On Shift have been added for users to modify logic when a vehicle goes on or off shift.

New EndRun Step

A new EndRun step has been added for use in the Processes window. The EndRun step may be used to set the ending time of the simulation run to the current time. This will cause the run to end once all simulation events scheduled for the current time have been processed.

Shadows within the View Ribbon

The Shadows icon within the View ribbon controls whether or not the shadows created by each object in the model are displayed. The light source in Simio is fixed in the corner at a 45 degree angle. However, if shadows are used in conjunction with a dynamic Skybox and the Day/Night cycle, the shadows will be adjusted for the change in the light source. Shadows were also added to the ThemePark example.

Functions for Entities

- **Sequence.CurrentJobStep** – Returns the current row index into the entity's assigned sequence table.
- **Sequence.NumberJobSteps** – Returns the total number of rows (i.e., job steps) in the entity's assigned sequence table.

Cost Multiplier support for Schedules

A *Cost Multiplier* property has been added to Work Schedules within the Day Patterns definition area. This multiplier is applied to the idle and usage cost rates of a resource following the work schedule specified. The *Cost Multiplier* property can also be found within the Work Period Exceptions when defining a GeneralException.

New SimBit

We have added a new SimBit to our SimBit library:

- **UsingAddRowAndOutputTable_Enterprise.spfx** demonstrates how to use the AddRow step in conjunction with Output tables. This SimBit applies to Simio Enterprise edition, which includes Output tables. Simulation data is written out to a table at the end of a simulation run, including entity entry and exit times from the system, as well as which server the entity has been processed on.

Replication Runner (Enterprise and Team Editions)

Simio Replication Runner is a program that allows an Experiment in Simio to distribute its replications to other computers on the local network. When other network computers are running Simio Replication Runner, the Simio experiment will use the resources on that computer for the experiment run, thus distributing the processing requirements across multiple computers on the network.

Resource Plan Gantt Chart Changes (Enterprise Edition)

Within the Resource Plan Gantt chart within the Enterprise edition, we've modified the presentation of Resource States, as well as Exceptions to be easier to view as well as edit.

Additionally, when a specific entity at a given resource is highlighted within the Resource Plan's Gantt chart, the table properties of that entity are now displayed within the properties window on the right. Any editable properties can then be changed directly within the Gantt view without changing to the Data Tables window.

Simio Release 4 – Sprint 65 – January 11, 2012

In this sprint we continue to enhance features within our product based on user feedback. Some of this sprint's changes include usage time based failures for resource, vehicle enhancements, new SimBits, and a number of new functions that have been added to provide the user with greater modeling flexibility and use of object references.

Usage Based Failures for Resources

Resources now have the usage time based failures within the Reliability section of properties. This provides greater functionality for resources, allowing downtimes to be based on not just calendar time, but alternatively based on the amount of time that the resource is actually in use.

HomeNode Variable for Worker / Vehicle

A new HomeNode variable has been added for users to dynamically assign the home location for a given worker/vehicle, using either Vehicle.HomeNode or VehicleName[x].HomeNode. The Initial Home location of both workers and vehicles can be changed during the simulation run using an Assign step.

Day/Night Icon for SkyBox

With dynamic skybox views in 3D, there is now a new day/night icon that can be turned off/on to display ambient light and a moving sun/moon.

Functions for Resource-enabled objects

- **Capacity.CurrentOwners** – The objects that currently own (have seized) capacity units of this resource.
- **Capacity.CurrentOwners.NumberItems** – Returns the number of objects that currently own (have seized) capacity units of this resource.
- **Capacity.CurrentOwners.FirstItem** - Returns a reference to the first owner in the list of owners that have currently seized capacity units of this resource.
- **Capacity.CurrentOwners.LastItem** - Returns a reference to the last owner in the list of owners that have currently seized capacity units of this resource.
- **Capacity.CurrentOwners.ItemAtIndex(index)** - Returns a reference to the owner at a specified index position in the list of owners that have currently seized capacity units of this resource.
- **Capacity.CurrentOwners.IndexOfItem(owner)** - Returns the one-based index of the first occurrence of a specified owner in the list of owners that have currently seized capacity units of this resource. If the owner has not seized the resource then the value 0 is returned.
- **Capacity.AllocatedTo(owner)** - Returns the current number of capacity units of this resource that are allocated to (have been seized by) a specified owner object.

DateTime Functions

We've added new functions that return a number representing a minute, hour, day, month, year, etc. based on a datetime input. There are also additional functions that convert a datetime to/from a string and those that let you determine which day of the week, month or year corresponds to a given datetime. These functions include:

- **Year, Month, Hour, Minute, Second**
- **DayOfYear, DayOfMonth, DayOfWeek**
- **DaysInMonth**
- **FromString, ToString**

Activity Element Functions

- **ActualDurationFor(entity)** - Returns the actual per batch time duration of this activity for the specified entity. Note that if the activity has not yet been started by the entity, then the duration returned will be based on if the entity was next to start the activity, respecting a sequence-dependent or change-dependent duration type accordingly.
- **BatchesRequiredFor(entity)** - Returns the total number of sequential batches required for the specified entity to perform the activity as per the operation quantity and activity batch size parameters.

Entity Object Functions

- **CurrentOperation** – If the entity object is currently performing an operation, then this function returns a reference to that operation.
- **CurrentActivity** – If the entity object is currently performing an operation activity, then this function returns a reference to that activity.

Workstation Functions

- **ActualSetupTime** – This function may be used as the dynamic selection expression to select the next job at a workstation, or in the selection expression to select a workstation from a list of candidate workstations at a node. It returns the actual setup time duration that would be applied if the associated entity was next to start operation on the workstation, respecting a sequence-dependent or change-dependent duration type accordingly.

Operation Element Functions

- **EstimatedMakespanFor(entity)** - Returns the estimated amount of time for the specified entity to perform this operation, from the start of the first activity to the end of the last activity, assuming the operation's activities will be spanned over (i.e., extended by) any off-shift periods for the operation's primary resource.

New SimBits

We have added three new SimBits to our SimBit library, including:

- **InitializeObjectPropertiesFromATable.spfx** demonstrates how to store and reference object specific data within a table. In this example, information about two servers is stored in different rows in a table. The *Auto-set Table Row Reference* property of an Object Reference column is used to point to the appropriate row.
- **RequestRideFromSameTransporter.spfx** shows how to request the same worker from a 'list' of workers that an entity had utilized earlier in the model for a task later in the simulation. The function `Entity.CurrentTransporter` is utilized to store information with the entity for later use.
- **UpdateStateInModelFromObject.spfx** demonstrates how a user-defined object can update the state of a model from within the object. The communication between objects in hierarchy is the focus of this example, which is an enhancement of the SimBit 'ProcessModelWithinModel.spfx'.

Performance Slider for Targets (Enterprise Edition)

Within the Targets / Tables / Gantt ribbons in the Enterprise edition, we've added a performance slider to easily allow users to change the upper and lower limits of target performance levels.

Simio Release 4 – Sprint 64 – December 1, 2011

There are two major components to this sprint. First we are releasing the long anticipated **Simio Enterprise Edition** with its Risk-based Planning and Scheduling (RPS) features. This culminates over a year of implementation. While we will continue to enhance the Enterprise feature set (just as we continue to enhance Design Edition), the general release of Enterprise Edition is an important milestone. Second, our “routine” enhancements to the base simulation products continue. Features include resource costing for Workstation objects and added material costing, ‘Random’ selection rule for selecting between resources, added user customized functions, and more.

Simio Enterprise Edition

Simio Enterprise adds a powerful set of patent-pending features. You can use Enterprise as either a richer, more feature-packed, environment for general modeling, or you can use it to extend your simulation models into daily operational support. Extend the use and lifespan of your previously built Simio models or build and execute new models for Risk-based Planning and Scheduling (RPS). Custom-tailor reports, graphs, and tables for use by schedulers. Reduce your risk and costs by analyzing your schedules in ways never before possible.

Enterprise features include:

Table States

Enterprise users can add State columns to a Data Table, which allows them to update our Data Tables with information. This updating can happen during the model run and then the contents of the Data Table can be exported after the run. This powerful feature provides users with a new method for keeping track of data during a run and also opens possibilities for getting run time data out of Simio into other production systems.

Unlike other Table column types, the table state column values are not inputted by the user at the start of the simulation run. These values are assigned during the simulation run by an Assign step. Table States may also be utilized in an expression. Just as you can define different types of states for an object, such as an Integer State or a String State, you can also define different types of States for a Table column. Other examples include Boolean, List, Object Reference and Level states, to name a few.

Targets

Enterprise users can add Targets to a Data Table to use as another type of output measure. A Target is an expression that will be evaluated at the end of the simulation run to determine if a particular goal is met, such as “Is this target within a user-defined acceptable range?”

A target’s Expression property is evaluated at the end of a simulation run and classified according to the upper and lower bounds specified. The user defines these Upper and Lower Bounds and then also specifies what text should be written to the table if the final value falls below the Lower Bound, above the Upper Bound or falls within the Bounds. An example might be to have the column display “On-Time” for orders that fall within Bounds or “Late” for orders that fall above the Upper Bound.

Custom-design a Scheduling Environment

You can define what the scheduling user can see and edit in the Data Tables. The Enterprise user can decide if a scheduling user (using the Scheduling Edition software available soon) or someone accessing the Tables from the Planning tab can view or edit each column of each table. The Enterprise model builder might want to allow other users to see certain columns of a table, not see other columns or even allow editing of a few columns. This is controlled by two properties on a table column called Visible and Editable found under the Operational Planning property category.

Resource Gantt Chart

Simio Enterprise Edition provides a Resource Gantt chart that graphically illustrates the states of constrained resources over time. It displays each entity that utilizes the resource. Within this Resource Gantt, you can graphically edit the resource's capacity, allowing you to add exceptions to a Work Schedule, such as adding overtime for a resource. The chart also allows the user to see which entities are waiting for the resource.

Entity Gantt Chart

Simio Enterprise Edition provides an Entity Gantt chart that graphically illustrates an entity's usage of resources over time. The user can also see each constraint that impedes the progress of the each entity. This Gantt also displays date-time based milestones and targets for each entity, along with any associated risk measures.

Detailed Logs

Resource Usage Log

This table provides detailed information on each resource and each entity that utilized it. Information such as the start time, end time, duration of use are available for a thorough analysis. This chart can be filtered and sorted.

Resource State Log

This log provides details of the resource states over time. It shows all the resources, their resource state (i.e. Busy, Failed, OffShift), when that resource entered that state and when it left that state. This chart can be filtered and sorted.

Constraint Log

This table shows a record of each constraint that impedes the progress of each entity. It aids in identifying bottlenecks and determining how to best improve a facility's operation. This table will indicate if an entity is being constrained because it is waiting on the capacity of a resource to become available or because it is waiting for an arrival of a resource, such as a Worker or Vehicle. The name of the constraint item is listed, along with a description of that item and when the wait time began and ended. This chart can be filtered and sorted.

Results and Analysis

Target Summary provides a high level summary of Target performance. There are also Risk Plots available for a graphical analysis of Target performance. After the Risk Analysis is run, the Risk Plots provide confidence intervals and percentiles for the target of each entity. Detailed Results, such as

queue lengths and wait times, are available in a pivot grid that can be filter, sorted or exported. This is similar to the Results pivot grid in a Simio model 's Results tab.

There are three reports available to display results of a Plan:

The *Resource Dispatch List* report includes chart and graphical representations for tasks on a particular resource for a given time period. The user inputs the specific Resource for which to display information and the report will display the entity, the start time, the end time and duration that this entity used this resource.

The *Workflow Constraint Analysis* report provides additional information to the Constraint Log within the Entity Workflow window. For each constraint type, the report includes information on the Constraint Item, number of Occurrences, Average Wait and Total Wait times, both in chart and graph formats. Constraints are then also displayed graphically by Facility location.

The *Resource Utilization Summary* report displays all the Resource states of each Resource and the total time it spent in that state. The information can be displayed in both a chart and graphically within the report.

General Simio Features

Random Selection Rule for Seizing Resources

We have added a new rule for selecting from Resource Lists, the 'Random' selection logic. Now, when seizing a resource / worker from a list of objects, you may randomly select between them. This is an enhancement that has been requested from many of our users.

Named Views for Pivot Grids

We have enhanced our Pivot Grid ribbon to include the Add View, Change View and Manage Views options, which users currently have within the View ribbon for saving animation views. However, these various views within the pivot grid allow users to save changes to sorting, filtering, column ordering and column width. Pivot Grids are available within the Results tab for running a simulation, experiment viewing when running multiple simulation replications and within the detailed results area of the Planning tab within Simio Enterprise edition for planning and risk analysis.

Costing for Materials and Workstation Objects

We have enhanced our costing features to include both material costs and workstation costs. The Material element now includes a Cost per Unit that is charged if a quantity of material is consumed. The Consume step includes the option to accrue material costs and a new Material element function, `MaterialCostCharged`, has been added.

Within the Workstation object of the Standard Library, resource costs have been added that include the Idle Cost Rate, Cost per Use and Setup, Processing and Teardown Cost Rates.

New Costing Functions

Two new functions are available for all intelligent objects that are considered resources, including Capacity.IdleCost and Capacity.UsageCostCharged. These functions will return the Idle Cost and Usage Cost Charged data items for a resource. This information is reported within the results of the simulation run.

User Defined Functions

We have added the ability for users to define their own custom functions within Simio. This can be done within the Definitions tab / Functions panel by selecting the **Function** button. A customized function may include a combination of other functions and/or expressions that the user is interested in viewing or analyzing during the simulation run.

This new capability supplements the previously available Function Tables on the Data tab which are useful for calculating a return value based on interpolating from a set of values. The new Functions allow you to create your own function name that will return the result of any expression you provide. This is particularly useful when you repeatedly use a long expression and you would like to instead reference that expression by a simple name.

Function Tables within the Data tab have been renamed **Lookup Tables** to avoid confusion and the keyword **Function** is no longer needed in front of the Lookup Table name.

Simio Release 4 – Sprint 63 – November 16, 2011

This sprint we continue to enhance the Risk-based Planning and Scheduling (RPS) features of Simio Enterprise Edition. We have also added resource costing to the Standard Library objects and made some UI changes. Our biggest feature in this sprint, however, is the addition of 'table-based' elements through the auto-create features. Now, within a data table, elements and their properties may be automatically created. This opens up many possibilities for importing bill of materials and so much more into the simulation model using tables and auto-create.

Data Binding Enhancements for Table Importing

There are some new enhancements to the data binding features within Simio, including the addition of database binding (i.e., binding a data table to Access, ODBC, Ole DB, Oracle or SQL Server). We've also added Binding Options, so that data within a table can be automatically updated / bound to a table at the start of a simulation run or manually imported.

Auto-Create Elements from Tables

Simio now has the capability of defining Elements and their properties from within Data Tables. This powerful feature enables users to store information, such as bill of materials and many other data items within Excel or other data sources, import into a table and automatically create various Elements.

For any given Element Reference column in a data table, the Reference Type property can be changed from the default value of 'Reference' (which will simply reference an existing element) to 'Create' (which will actually auto-create an element). Each element type has its own properties that can also be populated with a single value, expression or reference to another column in the data table.

Skybox Animation Feature

Within the 3D view of the Facility window, users now have the capability to define a SkyBox, which will place either a static or dynamic background or sky view within the animation. Dynamic options include clouds moving (typically the speed factor should be set to 5-10 or greater to see movement), while static options include various outdoor scenes. Users can create their own SkyBoxes as well. The "library" skyboxes are simply *.zip files with images inside them with the names front, back, left, right, top, bottom. They don't need to have all of those names in them, Simio will use whatever ones it finds. They are located in the Skybox folder under \Public\PublicDocuments\Simio.

SetRow Step Enhancement

The SetRow step within the Processes window has been changed to allow the token to assign a Table Name and Row Number to a specific object or element. This would include object / element reference names as well. This allows the user greater flexibility in assigning table data to specific objects, such as a Server for example.

Costing Enhancements

Resource Cost categories have been added to the Standard Library objects that are utilized as resources. This includes Server, Combiner, Separator, Resource, Vehicle and Worker objects. Resource costs include Idle Cost Rate, Cost Per-Use and Usage Cost Rate. Cost rates are accrued to the entity or owner object for the duration of the idle or busy time.

The Seize step now includes two new costing related properties within the Resource Seizes repeat group. These include the Accrue Usage Costs (True/False) and Per-Use Cost Accrual (At Seize, At Release, Next Station). These allow the user additional flexibility in determining whether a particular seize step will cause costing to be accrued and if so, when the per-use cost will occur.

Simio Enterprise Documentation

While the Simio Enterprise Edition has been available to early adopters for about 10 months, it will be made publicly available very soon. Enterprise may be thought of in two ways:

A new high-end version of Simio containing many valuable modeling features like:

- Specify states that can be added to a table. Table states can be written to during a run, thus updating your tables with output information.
- Add Targets to tables to use as another type of output measure: “Is a target within a user-defined acceptable range?”
- Model “builder” can customize tables for a model “user” controlling what is displayed and what of that can be changed.
- Resource Gantt chart to graphically illustrate states of constrained resources over time
- Entity Gantt chart to graphically illustrate an entity’s usage of resources over time
- Detailed Resource Usage Logs that list how long and when a Resource was used by a particular entity.
- Resource State Logs that summarize how long and when a Resource was in each state (for example; Busy, OffShift or Failed).
- Analysis of Targets both deterministic (plan) and stochastic (risk analysis).
- A Constraint Log which aids in identifying bottlenecks and determining how best to improve a facility’s operation.

A new tool for Risk-based planning and scheduling

- A very flexible modeling and analysis environment including all of the tools mentioned above.
- Works with Simio Scheduling Edition for deployment of a customized tool for schedulers.

If you are interested in learning more about Simio Enterprise Edition, both an introduction to Risk-based Planning and Scheduling (RPS), as well as the user’s guide for the new features are now included in the Simio on-line Help. Look for the major topic named “Scheduling in Simio Enterprise” or look for any of the sub-topics that include the word “Enterprise”. Please note that these features are only available in the Enterprise Edition, which requires an upgrade from Team Edition.

Simio Release 4 – Sprint 62 – October 25, 2011

This sprint, like the past several sprints, includes many new features in Risk-based Planning and Scheduling (RPS). We have enhanced our Costing capabilities, have added some features to the Processes window and continue to add SimBit examples to our extensive library.

Processes Window Ease of Use

The Processes window now has auto-scrolling capability, which will make navigating around larger processes much easier. When you want to move a step or a segment connector to another location within a large process, simply move the connector or step to the right or left and the window will auto-scroll. Additionally, the right-click menu for steps has been enhanced to provide options to Disconnect Exit and Disconnect Alternative Exit. These options are available if the step's exit can be disconnected without leaving part of the process disconnected from the Begin step.

Costing Enhancements

We've continued to add costing to the Simio product. With this sprint, Cost per Use and Holding Cost Rate properties have been added to every buffer within the Standard Library objects. These define costs to hold an entity in the buffer. For example, the Source includes these costs for its output buffer, while the Sink includes this costing information for its input buffer. Server, Workstation, Combiner and Separator now have buffer usage / holding costs for all input and output buffers.

Any transporter object, including the Vehicle & Worker objects in the Standard Library, now have a new Transport Costs section that define costs for an entity to use the transporter (cost per rider), as well as a transport cost rate.

The Station element now includes a 'Financials' section with Parent Cost Center, Cost per Use and Holding Cost Rate properties.

New Costing and Population Functions

In addition to costing properties within the Standard Library objects, there have been many new functions added for users to easily access costing information about the model, as well as population information about entities. These include:

For Queue States:

MinimumCostWaiting - Returns the minimum cost that an item accumulated while it waited in the queue.

MaximumCostWaiting - Returns the maximum cost that an item accumulated while it waited in the queue.

AverageCostWaiting - Returns the average cost that an item accumulated while it waited in the queue.

For entities / agents:

Population.CostPerItem.Average – Returns the average cost that was accumulated by destroyed objects of this object's type.

Population.CostPerItem.Minimum – Returns the minimum cost that was accumulated by destroyed objects of this object's type.

Population.CostPerItem.Maximum – Returns the maximum cost that was accumulated by destroyed objects of this object's type.

Population.Index – Returns the one-based index of this object in the population that the object is a member of.

Population.FirstItem – Returns the reference to the first member in the current of this object's type.

Population.LastItem – Returns the reference to the last member in the current of this object's type.

Population.ItemAtIndex(index) – Returns a reference to the member at a specific index position in the current population of this object's type.

Population.IndexofItem(object) – Returns the one-based member index of a specified object in the current population of this object's type. If the population does not contain the object, then the value '0' is returned.

New Costing SimBit

We have added a new Simbit to our extensive SimBit library:

- **Financials.spfx** highlights some of the new financial support recently added to Simio. The model demonstrates modeling Capital Cost, Buffer Holding Costs, Cost on an Entity and how to manually apply costs in process logic. The SimBit documentation contains detailed explanations for how the costs in the Results tab were calculated by Simio.

New Example

We have added **RSPExample.spfx** to the pre-installed examples. While this requires and is primarily intended for use with Simio Enterprise Edition, many of the basic modeling features are available for viewing with other Simio editions.

Simio Release 4 – Sprint 61 – October 3, 2011

This sprint includes many new features in Risk-based Planning and Scheduling (RPS) which is still in the early adopter phase. Since most people do not have access to those features they are not described here. (If you want more information, look to the Simio Products page or contact Simio for a personal demonstration). In addition we have improved code stability (that's marketing speak for fixed some bugs ☺). We have also added features that allow you to run larger models and make better use of your computer resources, as well as additional capabilities for Costing and Resources.

Experiment Replication Runner

Although Simio does not have any built-in memory limits, we were previously limited to the address space permitted by 32 bit operating systems. This limitation was most frustrating when you tried to take full advantage of multiple processors when running a large model. We have made two changes to address these problems.

- Multiple concurrent runs can now be made using separate processes. This will give everyone (regardless of your OS) the full memory available to your OS for use with each replication or scenario being run.
- If you are running a 64 bit OS, Simio can now take advantage of this when running experiments. Since 64 bit has the potential to address a huge memory space, each replication can use the full memory available to your system.

Under **Advanced Options** in **Experiment** you will find new options:

- **Replication Runner** – Provides 3 options to control how multiple replications are run.
 - **InProcess** runs replications in the same process as the main Simio executable. This is the fastest mode for small models, but is also the most limited in availability of memory.
 - **External32** takes advantage of separate processes for each run, but always uses 32 bit processing, even if you are on a 64 bit OS. This is a good compromise of making more memory available, while preserving execution speed.
 - **External** takes advantage of separate processes for each run, using either 32 bit or 64 bit based on your OS. On a 64 bit system, this may run some models slower, but can take full advantage of all available memory. Some very large models may require this setting.

Note, when you first use either of the external options, you may get a warning message from your firewall protection software which will result in an error in Simio. If you allow Simio access through the firewall (still internal to your computer), it should run normally.

- **Concurrent Replications** – Allows you to limit the number of replications that you will run concurrently on a multiprocessor system. This defaults to unlimited (0), but is appropriate when your model requires more memory than your current hardware and software configuration can provide.

Costing Features

Last sprint we introduced the Cost Centers element for the model, as well as Cost states for each object. This sprint, we have enhanced our costing features to now include Parent Cost Center and Capital Cost to most of the Standard Library objects, as well as Initial Cost and Initial Cost Rate for the ModelEntity. Costing calculations are then done automatically for each static object, such as Source, Sink, Server,

Workstation, Separator, Combiner, and Resource. These costing features are still at the early look phase – feel free to experiment with them and tell us what you think, but don't yet use them in a real project because they are still changing.

Resource Specific Exceptions

It is fairly common that facilities operate with a set of “standard” work schedules, but that specific people or devices might temporarily work a different schedule (e.g. “Joe, can you work 2 hours overtime on Wednesday?”). For many of our Standard Library objects that contain resources, we have added the ability to ‘override’ a specific schedule’s exceptions for that particular resource. For example, let’s say that Server1 follows an ‘AllServers’ schedule, which includes several work day and period exceptions (as defined in the Schedules panel of the Definitions window). Server1 may also specify a particular work day or work period exception that will ‘override’ those defined with the ‘AllServers’ schedule.

Simio Release 4 – Sprint 60 – September 12, 2011

Within this sprint of Simio 4, we've spent a lot of time adding new user functions and providing greater flexibility for users for modeling. We think you'll find the Object and Element reference states, in combination with the functions, very powerful in model building. Additionally, we've begun the process of adding Costing mechanisms to the software. This sprint provides you with some costing basics, allowing users to assign costs and specify currency types. Future sprints will expand on the costing basics to provide even greater functionality.

New Functions for Intelligent Objects

We have added many new functions to all intelligent objects (which include all Standard Library objects) as well as some specific functions for particular groups. Below is a list of those new functions:

Current Node - If the entity object's leading edge is currently at a node or the entity is located in a station owned by a node, then this function returns a reference to that node. Otherwise the Nothing keyword is returned.

DestinationNode - Returns a reference to the entity object's current destination node, or the Nothing keyword if the entity does not have a destination.

CurrentLink - If the entity object's leading edge is currently on or at the end of a link, then this function returns a reference to that link. Otherwise the Nothing keyword is returned.

TrailingLink - If the entity object's trailing edge is currently on or at the end of a link, then this function returns a reference to that link. Otherwise the Nothing keyword is returned.

CurrentStation - If the entity object is currently located in a station, then this function returns a reference to that station. Otherwise the Nothing keyword is returned.

BatchParent - If the entity object is currently located in a station, then this function returns a reference to that station. Otherwise the Nothing keyword is returned.

ReservedTransporter - If the entity object has made a reservation to ride on a transporter, then this function returns a reference to that transporter. Otherwise the Nothing keyword is returned.

CurrentTransporter - If the entity object is currently riding on a transporter, then this function returns a reference to that transporter. Otherwise the Nothing keyword is returned.

NetworkDistance.ToNode(node) - Returns the shortest travel distance from the entity object's current location to a specified node using the entity's current network.

NextEntityAheadOnLink - If the entity object's leading edge is on a link, then this function returns a reference to the next entity immediately ahead on the same link. If there is no entity ahead then the Nothing keyword is returned.

NextEntityBehindOnLink - If the entity object's trailing edge is on a link, then this function returns a reference to the next entity immediately behind on the same link. If there is no entity behind then the Nothing keyword is returned.

SeizedResources.NumberItems - Returns the current number of resources seized and owned by this object.

SeizedResources.FirstItem - Returns a reference to the first resource in the list of resources currently seized and owned by this object.

SeizedResources.LastItem - Returns a reference to the last resource in the list of resources currently seized and owned by this object.

SeizedResources.ItemAtIndex(index) - Returns a reference to the resource at a specified index position in the list of resources currently seized and owned by this object.

SeizedResources.CapacityUnitsOwned(resource) - Returns the number of capacity units of a specified resource that are currently seized and owned by this object.

Location.Parent - Returns a reference to the parent object that contains or is associated with the object's current location.

CurrentNetwork - Returns a reference to the entity object's currently assigned network.

Distance(fromNode, toNode) - Returns the shortest travel distance between two nodes using this network.

DirectDistanceTo.Object(object) - Returns the direct (straight-line) distance from this object to another specified object.

DirectDistanceTo.Location (x, y, z) - Returns the direct (straight-line) distance from this object to a specified coordinate location.

Population.Cost.Total – Returns the total cost that was accumulated by destroyed objects of this object's type.

Population.Cost.Average – Returns the average cost that was accumulated by destroyed objects of this object's type.

Population.Cost.Minimum – Returns the minimum cost that was accumulated by destroyed objects of this object's type.

Population.Cost.Maximum – Returns the maximum cost that was accumulated by destroyed objects of this object's type.

Object and Element Referencing

Object Reference and Element Reference state variables have been added to the Definitions tab, States panel of the product. This powerful feature allows users to store the names of elements and / or objects in state variables for use in various properties of objects, as well as functions. For example, the name of a Process can be stored within an 'Process' type of Element Reference state for an entity; then, within the model, the state name can be used when referencing a process. This would enable multiple entity types to reference different processes to be used within the same property field. This feature, along with the various functions that have been added, will allow users much more flexibility in specifying properties and enables 'macro' type modeling.

Costing Basics

The start of some basic costing constructs have been added to Simio, including a new Cost Center element, which can be found in the Elements panel of the Definitions tab. Cost Centers can be hierarchical by specifying parent cost center information. The Advanced Options under the Run ribbon now also includes several Financials options, including default currency and exchange rate values. Objects now include a 'Cost' level state variable, which permits the cost or cost rate to be changed by using the Assign step in the Processes window. There are multiple costing functions available as well.

Additional costing functionality, including automatic costing calculations within Standard Library objects, will be implemented in future sprints.

New SimBits

We have added two new SimBits to our SimBit library that demonstrate the object referencing states.

- **Object ReferenceOnEntity.spfx** demonstrates how one of the new functions, Location.Parent, can be used to store where an entity is currently located. This information is then later used in decision making logic.
- **SeizingSameResourceFromList.spfx** shows the use of an object reference state to store the name of the last seized resource that an entity was allocated (from a list in this case). The object reference state is then used in the next Server to seize the same resource, using the state name.

Importing tables into repeat groups

Table names and schema may now be converted from table format to repeat group format by using the 'Convert to Repeat Group' button in the Table ribbon. By selecting this, a table of the same name is created, with repeatable properties for each of the columns. The name of the table is then changed to be 'TableName_Original'. This feature allows the mapping of table data from a higher level in object hierarchy.

Persisting user settings

Users often customize the appearance of components in your model like column widths, table tab ordering, filtering on a particular column and setup of the categories within the Processes window. When a model is saved, your user-specified settings are now saved with the model so that when you open the model again, your changes are restored.

Load recent project

A 'Load most recent project at startup' button has been added to the Save window. If this box is checked, your project will automatically open when starting Simio.

Compatibility Issues

Objects with base classes who have newly introduced items with the same identifiers as items already in the object will cause less problems than they used to. Refer to the Known Issues.pdf file for more details.

Simio Release 4 – Sprint 59 – August 18, 2011

This past sprint includes several features that users have requested. We have also spent a good deal of time on object and element referencing, which should be available in the next sprint.

List Creation using Right Click in Facility Window

You can easily create your node, object, and transporter lists by simply right-clicking on the object(s) within the Facility window and adding them to (or deleting them from) an existing or new list. In previous Simio versions, a List was created by going to the Definitions window, Lists panel, specifying the list type and then entering all of the object names for that list. Now, you can individually or group select the items you'd like in a list and add them to an existing list or create a new list. Based on the type of objects selected, Simio determines the type of list that can be generated.

Create step and Separator object include Copy to a new Object Name

Prior to this sprint, when a copy of an entity was made using either the Create step or the Make Copies option of the Separator object, the copy was of the same type as the original entity. The option is now available to change the type of entity that is created. The information from the original entity is copied to the newly created entity, including state values, assigned network and destination, as well as table references. The size values, entity picture and any properties will be assigned from the new entity type.

Where's the Rest?

In case you are a history buff who would like to see a sprint by sprint account of how we got to this point, you can find that here:

<http://www.simio.com/downloads/public/documents/SimioHistoricalReleaseNotes.pdf>. This contains records from the first 59 sprints (0-58) covering the generation of Simio major releases 1 - 4.