GoldSim 11 Summary

Summary of Major New Features and Changes

July 2013





Table of Contents

Introduction	4
Documentation of New Features	4
Installation Instructions for this Version	4
Major New Features	4
Redesigned Simulation Settings	4
Controlling How Time History Results are Saved	6
Modifications to Run Properties That Can Be Referenced During a Simulation	7
Modeling Scenarios	8
New Result Display Features	12
Other Notable Modifications	18
New Dashboard Features	18
Naming Rules for Result Elements and Dashboards	20
Result Elements and Influences	20
Using Integrators to Compute Moving Averages	21
Support for Reverse Lookups, and Computing Integrals and Derivatives for Lookup Tables	21
New Triggering Events That Insert Unscheduled Updates	22
Referencing a Time Series Element Using a Function	23
Creating Dynamic Lookup Tables Using New Functions That Support Interpolated Lookups Within Vectors (1D-Arrays)	24
Improvements to Stochastic Elements	25
Customized Importance Sampling Using User-Defined Realization Weights	28
Support for Nested Monte Carlo Simulation	28
New Features to Facilitate Modeling Aging Chains	29
Minor Modifications	30
Improvements in How GoldSim Compresses and Stores Bitmap-Based Images	30
Modified Spreadsheet and Text Export from Time History Result Elements	30
Importing Units for Data from Spreadsheets	30
Support for .XLSM Files	31
Resetting an Extrema Element	31
Treatment of Fields with Numerator and Denominator Both Equal to Zero	31
New Features for Referencing Exposed Outputs of Localized Containers	32
Using Flags When Running Player Files from the Command Line	32
Modifications to Array Functions	33
	21

	Improvements to the Milestone Element	35
	Changes to Time Shifting Options for Time Series Elements	36
	Improvements to How Client Machines Access Floating Licenses	37
	Managing User-Defined Units	37
	Opening an Excel File and Selecting a Specified Sheet and/or Range via a Hyperlink Object	. 38
	Command Line Option for Global Database Download	. 38
	Reading in an Element Note from a Database	. 38
	Recording CPU Times for Elements	39
С	hanges to the Contaminant Transport Module	39
	Inclusion of ICRP Radionuclide Database in GoldSim	. 39
	Improvements to the Aquifer Pathway Element	40
	Improvements to the How Cell Pathways Manage Changing Media Amounts	41
С	onversion Issues When Reading and Running Existing Models	41
C	Conversion Issues When Reading and Running Existing Models	 41 42
С	Conversion Issues When Reading and Running Existing Models New Timestepping Options New Treatment of Run Properties	 41 42 42
С	Conversion Issues When Reading and Running Existing Models New Timestepping Options New Treatment of Run Properties Use of Keywords in Charts	 41 42 42 45
C	Conversion Issues When Reading and Running Existing Models New Timestepping Options New Treatment of Run Properties Use of Keywords in Charts Data Styles in Charts	 41 42 42 45 46
C	Conversion Issues When Reading and Running Existing Models New Timestepping Options New Treatment of Run Properties Use of Keywords in Charts Data Styles in Charts Change in How "Items" Unit Category is Defined	 41 42 42 45 46 46
C	Conversion Issues When Reading and Running Existing Models New Timestepping Options New Treatment of Run Properties Use of Keywords in Charts Data Styles in Charts Change in How "Items" Unit Category is Defined Log Interpolation of Independent Variables for Lookup Tables No Longer Supported	41 42 42 45 46 46 47
C	Conversion Issues When Reading and Running Existing Models New Timestepping Options New Treatment of Run Properties Use of Keywords in Charts Data Styles in Charts Change in How "Items" Unit Category is Defined Log Interpolation of Independent Variables for Lookup Tables No Longer Supported Changes to How SubModels Exchange Probabilistic Information with the Parent Model	41 42 42 45 46 46 47 47
C	Conversion Issues When Reading and Running Existing Models New Timestepping Options New Treatment of Run Properties Use of Keywords in Charts Data Styles in Charts Change in How "Items" Unit Category is Defined Log Interpolation of Independent Variables for Lookup Tables No Longer Supported Changes to How SubModels Exchange Probabilistic Information with the Parent Model SubModels No Longer Allow You to Define Number of Timesteps on the Input Interface	41 42 42 45 46 46 47 47 49
C	Conversion Issues When Reading and Running Existing Models New Timestepping Options New Treatment of Run Properties Use of Keywords in Charts Data Styles in Charts Change in How "Items" Unit Category is Defined Log Interpolation of Independent Variables for Lookup Tables No Longer Supported Changes to How SubModels Exchange Probabilistic Information with the Parent Model SubModels No Longer Allow You to Define Number of Timesteps on the Input Interface Changes to How Time History Result Elements Export to Spreadsheet and Text Files	 41 42 42 45 45 46 46 47 47 49
C	Conversion Issues When Reading and Running Existing Models New Timestepping Options New Treatment of Run Properties Use of Keywords in Charts Data Styles in Charts Change in How "Items" Unit Category is Defined Conger in How "Items" Unit Category is Defined Conger Supported Changes to How SubModels Exchange Probabilistic Information with the Parent Model SubModels No Longer Allow You to Define Number of Timesteps on the Input Interface Changes to How Time History Result Elements Export to Spreadsheet and Text Files Changes to Input Requirements for Event Delays and Timed Events Defined Using Stochastics	41 42 42 45 45 45 45 45 45 45 45 45 45 45 42 42 42 42 42 42 42 42 42 45 45 46 45 46 46 45 46 46 46 46 46 46 46 46 46 46 46 46 46 46 46 46 47 47 46 47
C	Conversion Issues When Reading and Running Existing Models New Timestepping Options New Treatment of Run Properties Use of Keywords in Charts Data Styles in Charts Change in How "Items" Unit Category is Defined Log Interpolation of Independent Variables for Lookup Tables No Longer Supported Changes to How SubModels Exchange Probabilistic Information with the Parent Model SubModels No Longer Allow You to Define Number of Timesteps on the Input Interface Changes to How Time History Result Elements Export to Spreadsheet and Text Files Changes to Input Requirements for Event Delays and Timed Events Defined Using Stochastics Changes in Advanced Timestep Settings	41 42 42 45 46 46 46 47 47 49 49 49 49 50

INTRODUCTION

This document describes the changes and new features implemented in GoldSim 11. This version includes a significant number of new features and user interface improvements. It also includes a number of bug fixes.

Documentation of New Features

This document summarizes the major changes in GoldSim11. All new features discussed in this document are described in detail in the accompanying Help file and user manuals for this release.

Of course, if you have questions about any of the new features in GoldSim, please contact us at <u>support@goldsim.com</u>. We also encourage you to visit our online user forum at <u>http://www.goldsim.com/Forum</u>.

Installation Instructions for this Version

You do not need to uninstall other GoldSim versions (such as GoldSim 10.11 or 10.50) in order to install GoldSim 11. In fact, until you have fully converted your existing models we recommend that you keep your previous GoldSim version installed on your machine. The new version will be installed in parallel to any existing pre-Version 11 GoldSim versions currently on your machine (it will not overwrite them).

To install GoldSim 11, you must download the full installation file: "GoldSim_11_Setup.exe". Download the file from the website, and run it. You should have at least 200 MB of disk space available before downloading.

After installing this version, when you start the program it will automatically detect and use your existing license. Note that you must have administrative privileges in order to install GoldSim successfully.

MAJOR NEW FEATURES

Redesigned Simulation Settings

The Time tab of the Simulation Settings dialog has been completely redesigned in GoldSim 11 to provide more intuitive and powerful tools for defining the timestepping options. The new dialog is shown below:

Ċ	Speary t	umestepping o	options for	the model.		
Bas	ic Settinas			Shor	w Schedu	led Updates
Tin	ne <u>B</u> asis:	Elapsed Tim	e 🔻	Time Displa	ay <u>U</u> nits:	day 👻
Du	ration:	100 day				
Sta	art Time:	3/25/2013		12:00:00 AM		
En	d Time:	7/ 3/2013		12:00:00 AM	A	
Alig	gnment:	Start Time a	aligned	▼ 1 day		
Ba	sic <u>S</u> tep:	User-specif	ied	▼ I day		200. T
<u>R</u> e	None	s:		Perioa:	Minor	Period;
U		Period Labe	; Major		Minor	r
Sa	ve Results:	Basic Steps		▼ Save en	very 1	Basic Step
(j):	101 schedule	ed update time	es, 101 sav	ved		

The key new features are as follows:

- Instead of specifying the number of timesteps, in GoldSim 11, you specify the timestep length. The final timestep is then adjusted to fit within the specified duration or start time/end time.
- For Calendar Time simulations, you can directly specify calendar-based timesteps (e.g., monthly, weekly). You can also control how these calendar-based timesteps are aligned (e.g., they can be aligned to start at the beginning of the month, on the 15th of every month, etc.).
- You can now create *Reporting Periods*, which compute and report *accumulated*, *average*, the *change* or the *rate of change* values over a specified period (e.g., monthly, annually). Reporting Periods can be used in conjunction with a shorter "Basic Step" (e.g., you may model the movement of water or money through a system using a daily timestep, but need to report the cumulative amount of money or water that moved from one point to another each month). In other cases, Reporting Periods may be the only timesteps you will need to specify.
- The concept of Phases has been replaced by two new features (defined via the **Advanced...** button): *Period Timesteps* (in which you define time periods where you would like to have shorter timesteps), and *Capture Points* (in which you specify points in time where Final Value-type results are captured).
- You can specify which day represents the first day of the week, and this information is used to align weekly timesteps, compute weekly reporting period results, and determine the value for the DayOfWeek Run Property.

- You can specify which month represents the first month of the year, and this information is used to compute annual and quarterly reporting period results.
- A button is provided that lists all the scheduled timesteps that will be used in your model based on the specified simulation settings. This provides a quick and easy way to understand where your timesteps will be.
- In previous versions of GoldSim, you could control whether or not "unscheduled timesteps" were inserted (e.g., to better represent events that fell between scheduled timesteps). By default, these were always inserted, but you could turn them off. In GoldSim 11, you can no longer control this. If your model is currently set to not allow unscheduled time steps, some model results will likely be different in GoldSim 11. "Unscheduled timesteps" are always inserted when required. Note, however, that a new option in Time History Result elements now allows you to view all timesteps in a plot (scheduled and unscheduled).

Because these changes are so extensive, it is highly recommended that you read the User's Guide or Help file to familiarize yourself with these new powerful features. These new capabilities are described in detail in the GoldSim User's Guide, starting on page 409. Alternatively, open the Help file, and on the Index tab enter *Simulation settings*, and then select the subtopic *time options*.

Controlling How Time History Results are Saved

A GoldSim model has the potential to produce a large amount of results. GoldSim does not produce separate input and output files. The model definition and the results generated by the model are all stored in one file: the gsm (<u>GoldSim Model</u>) file. Hence, for large models, particularly those with many timesteps and/or multiple realizations, when in Result Mode, your model file could potentially become very large.

Therefore, Time History Results and the Simulation Settings have been modified to prevent unnecessarily creating large result files. The most noticeable change in this regard is that for multiple realization (Monte Carlo) runs, time history results are only saved for outputs that are connected to Time History Result elements.

In addition, in previous versions of GoldSim, you could globally turn on and off the save flags for elements inside a Container. In GoldSim 11, you can turn off all the flags, but cannot globally turn them on. The option to turn them off is located on the Information tab of the Container element.

This is described in detail in the GoldSim User's Guide, starting on page 448. Alternatively, open the Help file, and on the Index tab enter *Results*, and then select the subtopic *specifying how saved*.

Modifications to Run Properties That Can Be Referenced During a Simulation

GoldSim 11 includes a complete redesign of the Run Properties (e.g., ETime, DayOf Week, etc.) to provide more intuitive and powerful tools for referencing these important properties.

The first thing you will notice is that the Run Properties are now organized into four categories (Calendar Time, Elapsed Time, Reporting Periods and Simulation):



Expanding any particular category folder lists all of the Run Properties in that category:

rch Options	Previous	Hom
🕰 Run Properties		
🔄 🕰 Calendar Time		
- 8 DateTime		
🔊 Year		
Quarter		
🔊 Month		
		- 11
···· 🖓 YearOfWeek		
🔊 DayOfMonth		Ξ
🔊 DayOfWeek		
R ⁰ Hour		- 11
🔊 Minute		- 11
^(*) Second		- 11
👘 DaysInMonth		
(1) StartTime		- 11
Elapsed Time		
Reporting Periods		
Simulation		2

The major modifications to Run Properties can be summarized as follows:

• In previous versions of GoldSim, some Run Properties (e.g., Time) had different meanings depending on whether you were running an Elapsed Time or Calendar

Time simulation. In GoldSim 11, this is no longer the case. All Run Properties have the same definition regardless of the time basis for the simulation.

In particular, all of the Calendar Time Run Properties (e.g., Month, DayOfMonth) exist even when running an Elapsed Time simulation; this is possible because a Start Time is still specified for Elapsed Time simulations, and this is used to compute all of the calendar-based Run Properties. Elapsed Time Run Properties (e.g., the elapsed day or elapsed year of a simulation) are referenced using unique names (e.g., EDay, EYear). For example, regardless of whether you are running an Elapsed Time or Calendar Time simulation, the calendar year and elapsed year can both be referenced: Year is the calendar year, and EYear is the elapsed number of years.

- Some of the Run Property names have been changed (and will be converted automatically when reading old models). For example, Time is now DateTime; Day is now either EDay (elapsed day of simulation) or DayOfMonth.
- A number of new Run Properties have been added, including RTime (the remaining time in a simulation) and several associated with the new Reporting Periods feature described above.

Because these changes are fairly extensive, it is highly recommended that you read the User's Guide or Help file to familiarize yourself with the new Run Properties. These new capabilities are described in detail in the GoldSim User's Guide, starting on page 440. Alternatively, open the Help file, and on the Index tab enter *Run properties*.

Modeling Scenarios

GoldSim 11 provides a powerful new capability that allows you to create, run and compare different scenarios for your model. Scenarios are differentiated by having different sets of input data. In particular, different scenarios have different values for one or more Data elements (referred to as Scenario Data). GoldSim's scenario modeling capability allows you to directly compare results generated by different sets of input parameters. In effect, when you use this capability, your model can store (and subsequently compare) multiple sets of outputs:



A probabilitic scenario result comparison (in which you compare distributions) would look like this:



A deterministic scenario result comparison would look like this:



This can be very useful for carrying out sensitivity analyses, testing and comparing alternative designs, and asking "what if" questions.

Scenarios are defined (and run) via a Scenario Manager dialog:

Scenario Name	Scenario Description Style	Show	Action
Base Case Scenario	Assumes average annual flow rates		Run Scenario
Low Flow Scenario	Flow rate is assumed to be low		Run Scenario
High Flow Scenario	Flow rate is assumed to be high		Run Scenario

Using this dialog, you can create new scenarios, and delete existing ones. When you add a new scenario, you are prompted for a Scenario Name and Description. In the example shown above, three scenarios have been defined.

Scenario Data are the Data elements that differentiate the various scenarios in your model. GoldSim provides several ways to specify Scenario Data. The most straightforward way to specify Scenario Data is to use the Scenario Manager, which provides a sub-dialog for specifying all of the Data elements defining the scenarios:

Element ID	Base Case Scenario	Low Flow Scenario	High Flow Scenario	Live Model	
Action:	Copy From	Copy From	Copy From	Copy From	
Capacity	100 m3	120 m3	120 m3	100 m3	
Pipeline_Discharge	500 m3/day	300 m3/day	700 m3/day	500 m3/day	
Stream_Inflow	1000 m3/day	500 m3/day	1200 m3/day	1000 m3/day	

Each row in this dialog represents a Scenario Data element. Each column represents a scenario. You can add new Scenario Data using the **Add Element...** button. You can subsequently edit the value for any Scenario Data element in any defined scenario. This table is very convenient because it displays all Scenario Data values for all defined scenarios side-by-side.

You can also edit Scenario Data directly in your model. Once you have defined at least one Scenario in your model, the dialog for Data elements is modified slightly. In particular, the dialog provides an option to define a Data element as a Scenario Data element:

Definition	
Element <u>I</u> D:	Stream_Inflow
Description:	
Display Units:	m3/day Type Scalar
	• • • • • • • • • • • • • • • • • • •
Data Definitio	n
1000 m3/d	ау
Editing is lim scenario dat Case Scenar	ay ted because this element provides a. Displayed data is for scenario 'Base io'.
Editing is lim scenario dat Case Scenar	ay ted because this element provides a. Displayed data is for scenario 'Base io'.
Editing is lim scenario dat Case Scenar Data Source	ay ted because this element provides a. Displayed data is for scenario 'Base io'. e be: None v
Editing is lim scenario dat Case Scenar Data Source Typ Save Result	ay ted because this element provides a. Displayed data is for scenario 'Base io'. S None
Editing is lim scenario dat Case Scenar Data Source Typ Save Result	ay ted because this element provides a. Displayed data is for scenario 'Base io'. S S S Final Values I Time Histories

Because scenario modeling is somewhat complex (as your model will be storing multiple sets of inputs and outputs), it is strongly recommended that you read the User's Guide or Help file to familiarize yourself with this powerful new feature. This new capability is described in detail in the GoldSim User's Guide, starting on page 458. Alternatively, open the Help file, and on the Index tab enter *Scenarios*, and then select the subtopic *overview*.

New Result Display Features

The entire result display framework has been updated in GoldSim 11. Although most of the new features and capabilities are self-explanatory (or can easily be understood by viewing context-sensitive Help), to fully take advantage of the new features, it is strongly recommended that you first read (or at least skim) the documentation section that discusses result display. This can be found in the GoldSim User's Guide, starting on page 505. Alternatively, open the Help file, and on the Index tab enter *Displaying results*, and then select the subtopic *introduction*.

Many of the changes will appear to be only cosmetic, but the entire underlying framework has been rewritten to make it more efficient and powerful.

Some of the most important new capabilities are summarized below.

Ability to Define and View Results Based on Reporting Periods

As mentioned above, you can now create *Reporting Periods*, and Time History elements can compute and report *accumulated*, *average*, the *change* or the *rate of change* values over a specified period (e.g., monthly, annually). Reporting Periods can be used in conjunction with a shorter "Basic Step" (e.g., you may model the movement of water or money through a system using a daily timestep, but need to report the cumulative amount of water or money that moved from one point to another each month).

Results based on Reporting Periods (e.g., cumulative values over each period, average values over each period, etc.) can be accessed and viewed via Time History Result elements.

For example, the chart below plots a daily flow rate along with the average flow rate for each monthly Reporting Period:



The chart below plots a daily flow rate along with the cumulative flow for each annual Reporting Period:



It should be noted that most spreadsheet models implicitly compute accumulated values (e.g., flows) over a step. Hence, if you want to compare or integrate GoldSim with a spreadsheet model, you can use Reporting Periods to define your steps, and view the cumulative result.

Reporting Periods are described in detail in the GoldSim User's Guide, starting on page 554. Alternatively, open the Help file, and on the Index tab enter *Reporting periods*, and then select the subtopic *viewing results*.

Viewing Unscheduled Updates (High Resolution Results)

In some cases, events or other changes in the model may not fall exactly on a scheduled update. That is, some events or changes may actually occur between scheduled updates of the model. These trigger an "unscheduled update" of the model. Unscheduled updates are timesteps that are dynamically inserted by GoldSim during the simulation in order to more accurately simulate the system. Unscheduled updates can no longer be disabled in the simulation settings.

Unlike scheduled updates, unscheduled updates do not normally appear in time history plots and tables. That is, although these timesteps may affect the results (e.g., by making them more accurate at the scheduled timesteps), unscheduled updates of the model are not saved and displayed. Only the scheduled updates are actually saved and displayed.

In some cases, however, it may be of interest to see the values of selected outputs at unscheduled updates (e.g. for model debugging). To facilitate this, GoldSim 11 provides an option in the Advanced Time settings to save results at unscheduled updates (referred to as "high resolution" results).

The following example shows a chart of an overflow rate plotted only at scheduled timesteps (in this case once every 10 days) superimposed with the same result showing unscheduled timesteps:



In this example, the Reservoir starts to overflow in the middle of a timestep (at about 43 days). This causes an unscheduled update to be inserted.

Displaying results at unscheduled timesteps is described in detail in the GoldSim User's Guide, starting on page 570. Alternatively, open the Help file, and on the Index tab enter *Unscheduled updates*, and then select the subtopic *viewing results*.

Ability to Define Custom Statistics When Viewing Multiple Time History Results

In some cases, you may want to view probabilistic results for multiple outputs (e.g., flow rate, volume), but would like to display different statistics for each (e.g., the mean flow rate and the 95^{th} percentile of the volume).

GoldSim 11 allows you to define and display a custom statistic for each output displayed in a chart or table:



Defining and displaying custom statistics is described in detail in the GoldSim User's Guide, starting on page 550. Alternatively, open the Help file, and on the Index tab enter *Time history results*, and then select the subtopic *viewing custom statistics*.

Viewing Results for Multiple Scenarios in a Single Display

As described above, GoldSim 11 provides a powerful new capability that allows you to create, run and compare different scenarios for your model. Scenarios are differentiated by having different sets of input data.

GoldSim's scenario modeling capability allows you to directly compare results generated by different sets of input parameters. In effect, when you use this capability, your model can store (and subsequently compare) multiple sets of outputs. In particular, result displays like time histories and distributions can be plotted for different scenarios:



User Control Over Labels Used in Legends, Titles and Headers

GoldSim 11 allows all results that are being plotted to be assigned custom labels (that can have spaces, unlike element and output names). These labels are subsequently used in legends and titles in charts, and headers in tables, making these result displays much more readable.

Classifying and Screening Results

When carrying out probabilistic simulations, you may often run hundreds or thousands of realizations. In order to analyze the results, GoldSim 11 allows you to classify the realizations into categories. A category is simply defined by a condition relating one or more outputs in the model (e.g., those realizations in which the discount rate was above 3.5%; those realizations in which the profit exceeded \$1,000,000; those realizations in which the peak concentration was between 1 mg/l and 10 mg/l).

The power of categories is that once you have defined them, you can use them in two ways:

- Within certain kinds of result charts (e.g., scatter plots, time histories, distributions), realizations from each category can be displayed in a different color (and/or symbol).
- Within all result displays, you can choose to screen out one or more categories, so that the results that are shown (in charts and tables) only include those realizations in the categories which you have chosen to include.

This capability is described in detail in the GoldSim User's Guide, starting on page 524. Alternatively, open the Help file, and on the Index tab enter *Classifying realizations*.

OTHER NOTABLE MODIFICATIONS

New Dashboard Features

Showing/Hiding Controls Dynamically in Dashboards

In some models, some Dashboard input controls (or output controls) may not be applicable based on the values of other inputs (or outputs) in the model. For these situations, GoldSim 11 allows you to disable or hide Dashboard controls based on various model attributes. (This feature was actually added in 10.5 SP1, but was not formally documented).

Disabling a control greys it out on the Dashboard (so it cannot be edited). When a control is hidden, it cannot be seen at all on the Dashboard (hiding a control also disables it).

Examples of what you can do with this feature include the following:

- Show a control only if a different checkbox control on the Dashboard is checked.
- Enable an input control only if another input control's current value is greater than 10.
- Show an output control only if a specific model output is True.
- Enable a control only if a specific model output's current value is between 0 and 100.
- Enable a control only if a specific model output's current value is one of several specified values.

This new capability is described in detail in the GoldSim Dashboard Authoring Module User's Guide, starting on page 77. Alternatively, open the Help file, and on the Index tab enter *Dashboards*, and then select the subtopic *disabling and hiding controls*.

New Button Command to Import All Outputs from Spreadsheet Elements

July 2013

GoldSim 11 provides a new Button command ("Refresh all Spreadsheet outputs") that imports all outputs from linked Spreadsheet elements into your model. This is particularly useful if you want to import and view outputs in a Dashboard prior to running simulation.

New Button Command to Export Results from Time History Result Elements

GoldSim 11 provides a new Button command ("Export Simulation Results") that executes a global export of available results from Time History Result elements that are specified to export to spreadsheets and/or text files. This is equivalent to pressing "Export Now" button from Results tab of Options dialog in GoldSim.

This is particularly useful if you want to Player users to be able to export scenario results after running multiple scenarios.

New Capability to Open an Excel File and Select a Specified Sheet and/or Range via a Dashboard Button

The Open File command for a Button control opens a specified file. Doing so launches the application associated with the specified file and opens the file in the application. For example, if you specified a Microsoft Excel file (*.xls, *.xlsx or *.xlsm), GoldSim would launch Excel and open the file.

When opening a spreadsheet file, in some cases you may want to immediately jump to a specified location (sheet and cell range) when opening the file. GoldSim 11 provides specialized syntax to support this. The syntax is as follows:

Filename#sheetname!cellrange (e.g., Book1.xlsx#Sheet1!a1:c3)

This new capability is described in detail in the GoldSim Dashboard Authoring Module User's Guide, starting on page 66. Alternatively, open the Help file, and on the Index tab enter *Spreadsheets*, and then select the subtopic *opening at specified location using a button*.

New Dashboard Output Control: Display a Text String Based on an Output Value

Previous versions of GoldSim provided a Dashboard control (the Result Status Display control) that allow you to display one of a series of icons based on the value of an output. GoldSim 11 provides a new control (the Text Result control) that allows you to display one of a series of user-specified text strings based on the value of an output (e.g., if the

value is less than 10, display "Low", if the value is between 10 and 30 display "Medium", and if the value is greater than 30 display "High").

This can be used to visually indicate the state of a variable in your Dashboard (e.g., whether a pump is on or off; whether a variable has reached a dangerous level). It behaves similarly to (and can be used in conjunction with) a Result Status Display control. That is, while a Result Status Display control displays different icons based on the value of an output; a Text Result control displays different text strings based on the value of an output.

This new Dashboard control is described in detail in the GoldSim Dashboard Authoring Module User's Guide, starting on page 59. Alternatively, open the Help file, and on the Index tab enter *Text result control*.

Selecting, Creating and Running Scenarios from a Dashboard

The new scenario feature is fully supported within Dashboards. A new Scenario control allows Player users to create, run and compare scenarios:



This new Dashboard control is described in detail in the GoldSim Dashboard Authoring Module User's Guide, starting on page 71. Alternatively, open the Help file, and on the Index tab enter *Scenario control*.

Navigating to Dashboards Within SubModels

Previous versions of GoldSim provided a Command within a Dashboard Button to jump to another Dashboard. However, if the Dashboard was contained within a SubModel, it was not accessible (i.e., you could not navigate to a Dashboard within a SubModel).

GoldSim 11 now allows you to navigate directly to a Dashboard within a SubModel.

Naming Rules for Result Elements and Dashboards

In GoldSim 11, unlike other elements in GoldSim, the names of Result elements and Dashboards can contain spaces. Like other elements, however, they still can only include letters, numbers and the underscore character.

Result Elements and Influences

In previous versions of GoldSim, if an output was referenced in a Result element, an influence was drawn from the output's element to the Result element. This often resulted

in very messy and busy Graphics Panes. In GoldSim 11, these influences are no longer drawn (i.e., influences are never drawn to Result elements).

Using Integrators to Compute Moving Averages

In some cases you may want to represent a moving average in your model. For example, at any given time, you might want to know what the average value was over the previous 30 days.

GoldSim 11 provides an option for an Integrator to compute up to three different moving averages. A moving average is a uniformly-weighted average of the Rate of Change over a specified averaging time. It is calculated as the change in the value of the Integrator over the averaging time divided by the averaging time.

This new capability is described in detail in the GoldSim User's Guide, starting on page 231. Alternatively, open the Help file, and on the Index tab enter *Moving averages*.

Support for Reverse Lookups, and Computing Integrals and Derivatives for Lookup Tables

GoldSim 11 provides four new functions that can be used to manipulate Lookup Tables. To implement these, you must add a second argument to the table function.

Perhaps the most common function you will use is the reverse lookup. For example, suppose that you had defined a Lookup Table in which the independent variable is the volume of water in a pond, and the dependent variable is the water elevation in the pond (hence, the values in the table are determined by the shape of the pond). You would then use the Table to return a value for the water elevation for a specified value of the volume. However, what if you wanted to know the corresponding volume for a specified value of the water elevation?

GoldSim now supports this by allowing you to carry out a reverse lookup; that is, you can reference a 1-D table by sending it a value for the *dependent variable*, and the function will return the corresponding value for the *independent variable*.

To implement this, you must add a second argument to the table function:

```
Tablename(Row Variable, TBL_Inverse)
```

Using the example described above, if you had a Lookup Table (named Elevation) in which the independent variable is the volume of water in a pond, and the dependent variable is the water elevation in the pond, then this expression would return the elevation corresponding to a volume of 150 m^3 :

Elevation(150 m3)

and this expression would return the volume corresponding to an elevation of 10 m:

Elevation(10m, TBL_Inverse)

Three other functions allow you to evaluate the derivative and integral of the function defined by a Lookup Table:

Tablename(Row Variable, TBL_Derivative): This returns the derivative of the function at the specified row value.

Tablename(Row Variable, TBL_Integral): This returns the integral of the table function up to the specified row value.

Tablename(Result Variable, TBL_Inv_Integral): This returns the row value corresponding to a specified integral for the table result (the result variable).

For example, if you had a Lookup Table (named VOL) in which the independent variable was a length and the dependent variable was a volume, then this expression would return the derivative of the corresponding function (an area) when the independent variable was 5 m:

Vol(5m, TBL_Derivative)

One practical application of the use of these functions is when creating tables that describe how the surface area, volume and water elevation in a pond or reservoir are related. In these kinds of problems, you often would compute the volume, and then use a table to output the surface area and elevation corresponding to that volume. One way to do this in a consistent way using the table functions would be to define a table (named, say, Area_from_Elevation) that outputs the surface area as a function of the water elevation (i.e., the independent variable would be elevation and the dependent variable would be surface area). Given a volume (V), you could then use this table to compute the elevation and surface area as follows:

Elevation = Area_from_Elevation(V, TBL_Inv_Integral)

Surface_Area = Area_from_Elevation(Elevation)

These new functions are described in detail in the GoldSim User's Guide, starting on page 280. Alternatively, open the Help file, and on the Index tab enter *Lookup Tables*, and then select the subtopic *inverse lookup* or *Derivative of Lookup Table*.

New Triggering Events That Insert Unscheduled Updates

GoldSim 11 includes 3 new types of triggers (that can be used in all triggering dialogs):

At Stock Test: The Trigger Definition must be a conditional expression of the form "A>B", "A>= B", "A<B", "A<=B", or "A=B" where A is a Reservoir (or amount of medium or level in a Cell element). The element is triggered whenever the Trigger Definition *becomes* True. An event is created at the exact point in time that this condition becomes true.

At Date: The Trigger Definition must be a date or date and time, enclosed in quotations. (Alternatively, the date can also be expressed as the amount of time since

30 December 1899). The element is triggered whenever the simulated date reaches the specified date and an event is created.

At ETime: The Trigger Definition must be an elapsed time. The element is triggered whenever the simulated elapsed time reaches the specified elapsed time and an event is created.

The significance of these new trigger types is that unlike *On True*, *On False* and *On Changed* triggers, these new triggers interrupt the clock (i.e., insert unscheduled updates) so that the events they correspond to can be represented very accurately in time.

To understand the implications of this, consider an example in which your scheduled updates (i.e., timesteps) were every 10 days. There are two different ways you could try to trigger an event when the value of Reservoir A became greater than the value B. You could create an *At Stock Test* trigger of A > B, or you could create an *On True* trigger of A > B. If we assume that A > B actually became true at 15 days, these two triggers would behave very differently. The *At Stock Test* trigger would catch this point exactly, and insert an unscheduled update at 15 days. In the absence of any other unscheduled updates, however, the *On True* trigger would not be evaluated and implemented until 20 days.

Similarly, if you wished to trigger an event at a specific elapsed time (e.g., 17 days), you could try to do so in two different ways. You could trigger the event using an *At ETime* trigger of 17 days, or you could create an *On True* trigger with ETime >=17days. Again, these two triggers would behave very differently. The *At ETime* trigger would catch this point exactly, and insert an unscheduled update at 17 days. In the absence of any other unscheduled updates, however, the *On True* trigger would not be evaluated and implemented until 20 days.

These new triggers are described in detail in the GoldSim User's Guide, starting on page 320. Alternatively, open the Help file, and on the Index tab enter *Triggering*, and then select the subtopic *defining triggering events*.

Referencing a Time Series Element Using a Function

In some situations, you might want to reference a value from a Time Series element at some point in the future or past (as opposed to at the current time in the simulation). To support this, Time Series elements can be referenced in the way that you would reference a built-in function (like sin or min). That is, once you define a Time Series, you can reference it in input expressions for other elements as if it were a function:

Equation Rainfall(35 day)

This expression instructs GoldSim to use the Time Series defined by the element named Rainfall and compute its value at time = 35 days.

Of course, once you have defined the table, you can reference it at multiple locations (i.e., in the input fields for various elements) using different input arguments (different time values) in the same manner as you would use the built-in functions provided by GoldSim multiple times.

The dimensions of the Time Series function are determined by the Display Units specified when the Time Series element was defined. Note that if the Time Series itself is defined as an array, the output of the function is an equivalent array. To access the value from the first item of the time series array, you have to write the function as follows: getrow(Rainfall(35 day), 1)

Time Series functions have two arguments (with the second one being optional). In most cases, you will call the function with only the first argument. The first argument is the time value. If you omit the second argument, the first argument is interpreted in the same manner in which the Time Series was defined. Hence, if the Time Series was defined based on Calendar Time, the first argument is assumed to be a date. If the Time Series was defined based on Elapsed time, the first argument is assumed to be an elapsed time. By using the second argument, you can specifically override the default setting for how the first argument in interpreted (e.g., you can force it to be interpreted as an elapsed time even if the Time Series was define based on Calender Time).

This new capability is described in detail in the GoldSim User's Guide, starting on page 201. Alternatively, open the Help file, and on the Index tab enter *Time Series elements*, and then select the subtopic *referencing as a function*.

Creating Dynamic Lookup Tables Using New Functions That Support Interpolated Lookups Within Vectors (1D-Arrays)

Two new array functions are available in GoldSim 11 (these were actually added in 10.5 SP1, but were not formally documented).

The primary purpose of these two functions (which will often be used together) is to allow you to create dynamic lookup tables (in which the values for the independent and/or dependent variables are changing with time). A Lookup Table element does not support dynamic tables (all the values must be constant values).

vIndex(vector, value) requires the first argument to be a vector, and the second argument to be a value (with the same dimensions as the vector). It searches for the specified value within the provided vector, and returns the dimensionless *index* at which the value is found. It treats the vector as a continuous function, linearly interpolating where necessary between the defined points in the vector, and as a result, the returned index value is not necessarily an integer. For example, if the vector's values were [3m, 4m, 5m, 6m], then vIndex(vector, 5.1m) would return an index value of 3.1. Similarly, vIndex(vector, 10ft) would return an index value of 1.048.

If the value is outside of the range of values in the vector, a fatal error is displayed. If the value exists multiple times in the vector (e.g., a sine function), the first matching value is used.

vInterp(vector, index) requires the first argument to be a vector, and the second argument to be a dimensionless index. It interpolates into the vector using the specified index value. It treats the vector as a continuous function, linearly interpolating where necessary between the defined values in the vector, and returns a value (with the same dimensions as the vector). For example, if the vector's values were [3m, 4m, 5m, 6m] then vInterp(vector, 3.1) would return a value of 5.1m.

If the index is outside of the range of indices for the vector, a fatal error is displayed.

These two functions can be used in conjunction to create a dynamic lookup table. To do so, you would create two vectors using the same array label set, which would typically be indexed (e.g., 1 through 10). One of the vectors (named VectorInd here) would contain the independent variables, and the other vector (named VectorDep here) would contain the corresponding dependent variables. The values of these could change dynamically. At any time, if you wanted to obtain the dependent variable for a particular value of the independent variable (say, X), you could do so using the following expression:

vInterp(VectorDep, vIndex(VectorInd, X))

This would look up the index of X in the vector representing the independent variable and return the corresponding value from the vector representing the dependent variable.

A reverse lookup into the same table could be carried out using this expression:

vInterp(VectorInd, vIndex(VectorDep, Y))

This would look up the location of Y in the vector representing the dependent variable and return the corresponding value from the vector representing the independent variable.

These new functions are described in detail in the GoldSim User's Guide, starting on page 722. Alternatively, open the Help file, and on the Index tab enter *Vectors*, and then select the subtopic *using as lookup tables*. In addition, an example file illustrating how to use these functions to implement a dynamic lookup table (Dynamic_LookupTable.gsm) can be found in the General Examples/LookupTable folder (accessed via **File | Open Example...** in the main menu).

Improvements to Stochastic Elements

GoldSim 11 includes several improvements to the Stochastic element.

New Distribution Types: BetaPERT and Log-Cumulative

A new distribution type called a BetaPERT was added. This is essentially a specialized Beta distribution that is defined by a minimum, most likely, and maximum value. The new distribution is described in detail in the GoldSim User's Guide, starting on page 162. Alternatively, open the Help file, and on the Index tab enter *BetaPERT Distribution*. The mathematical details of the distribution can be found in Appendix B of the User's Guide.

A new option was added to the Cumulative distribution to allow you to define the distribution as a log-cumulative. Whereas in a cumulative distribution, the density between values is constant (i.e., the distribution between values is uniform), in a log-cumulative, the density of the *log* of the value is constant (i.e., the distribution between values is log-uniform). Log-Cumulative distributions are often applied to quantities with large (order-of-magnitude) uncertainties. The mathematical details of the distribution can be found in Appendix B of the User's Guide.

Importing Distributions from Spreadsheets

You can now use the Spreadsheet element to import a probability distribution from a spreadsheet. The distribution is defined in the spreadsheet using a number of fields that specify the type of distribution, as well as the arguments to the distribution (the probability distributions must be specified in the spreadsheet in a very specific format).

Once the distribution is imported via the Spreadsheet element, it can be used to define a Stochastic in your model (using the new "Externally-Defined" distribution type).

This new feature is described in detail in the GoldSim User's Guide, starting on page 842. Alternatively, open the Help file, and on the Index tab enter *Spreadsheet elements*, and then select the subtopic *importing probability distributions*.

Dynamically Revising Distributions Using Simulated Bayesian Updating

GoldSim 11 provides an advanced feature for dynamically revising distributions during a simulation using "simulated Bayesian updating". This feature is most valuable when using GoldSim to simulate a project of some kind (e.g., developing a new product, building a new facility). It is often the case that as a project progresses, additional information about uncertain variables is acquired. From the project operator's point of view, their uncertainty is decreasing, and hence as time progresses, the probability distributions used in their model of the project are converging towards the true values of the variables. As their knowledge increases they are better able to make appropriate decisions for the project's path forward.

When simulating a project in this manner using GoldSim, the model that is constructed will use Stochastic elements to specify the initial probability distributions of uncertain parameters, and a Monte Carlo sampling process will be used to pick the 'true' values of

those parameters. Given that starting point, simulated Bayesian updating can be used to simulate the additional information that will accrue as the project progresses. This in turn can be used to simulate the decisions that would be made based on the new information.

This new feature is described in detail in the GoldSim User's Guide, starting on page 931. Alternatively, open the Help file, and on the Index tab enter *Simulated Bayesian updating*.

Database Download of Stochastics Can Modify the Distribution Type

Previous versions of GoldSim have been able to download the definition of a Stochastic element directly from a database (either a "Simple" database, or a "Yucca Mountain" database). However, prior to GoldSim 11, the distribution type specified in the Stochastic element had to match the type specified in the database (i.e., the database could import the parameters of a distribution, but could not change the distribution type).

In GoldSim 11, the database download can change the distribution type originally specified in the Stochastic element.

New Functions that Operate on Distributions

In GoldSim 11, Stochastic elements have an additional output (the Distribution definition). This output is a complex output representing all the distribution information. This output type can also be produced by a Spreadsheet element (see above; this is how a distribution can be imported from a spreadsheet) and a SubModel (discussed on page 47).

GoldSim provides several specialized functions that can be used to reference the Distribution output type and return properties of the distribution. For example, the function PDF_Mean(X) returns the mean of Distribution X; the function PDF_Value(X, 0.95) returns the 95th percentile.

Since these functions simply return the key properties of a probability distribution, in most cases, there would generally be no value in using these functions directly on Stochastic element outputs. However, there are two primary applications for these functions:

- When the definition of a Stochastic distribution is changing dynamically during a simulation (either by making the defining parameters functions of time, of via Bayesian updating, see below), these functions can be used to query the current shape and characteristics of the distribution.
- When exporting a distribution result from a Monte Carlo SubModel, these functions can be used to query the current shape and characteristics of the distribution in the parent model.

This new feature is described in detail in the GoldSim User's Guide, starting on page 181. Alternatively, open the Help file, and on the Index tab enter *Stochastic elements*, and then select the subtopic *distribution functions*.

Customized Importance Sampling Using User-Defined Realization Weights

For risk analyses, it is often necessary to evaluate the low-probability, high-consequence portions of the probability distribution of the performance of the system, and that can require carrying out a very large number of Monte Carlo realizations. Because the models for such systems are often complex (and hence need significant computer time to simulate), it can be difficult to use the conventional Monte Carlo for such risk analyses.

To facilitate this type of analysis, GoldSim allows you to utilize an importance sampling algorithm to modify the conventional Monte Carlo approach so that selected portions of input distributions (which could correspond to high-consequence, low-probability outcomes) are sampled with an enhanced frequency. During the analysis of the results that are generated, the biasing effects of the importance sampling are reversed. The result is high-resolution development of "high-consequence, low-probability outcomes", without paying a high computational price.

Stochastic elements provide built-in functionality to force importance sampling on either the low end or high end of a Stochastic element's range (i.e., the tails of distribution). Timed Event and Random Choice elements provide similar functionality.

In some cases, however, it may be necessary apply importance sampling not just over the tails, but over user-defined regions of the Stochastic element's range. For example, if it was known that the important results of a model were highly sensitive to values of a particular Stochastic that were close to the 40th percentile, you would want to focus sampling in that region.

GoldSim 11 supports this by providing three specialized functions that provide biased sampling (and compute the appropriate weights) for a targeted range within a probability distribution. All three functions require a Target probability and a Width. For example, if you specify a Target of 0.4 and a Width of 0.05, biased importance sampling will be focused in the region between probability level 0.375 and 0.425. By using these functions in conjunction with a new capability in GoldSim 11 by which the user can manually specify realization weights, a customized importance sampling scheme can be implemented.

This new advanced feature is described in detail in the GoldSim User's Guide, starting on page 928. Alternatively, open the Help file, and on the Index tab enter *Importance sampling*, and then select the subtopic *custom*.

Support for Nested Monte Carlo Simulation

Many models have uncertain (epistemic) parameters as well as randomly variable (aleatory) parameters. An uncertain parameter represents ignorance that can theoretically

(but perhaps not practically) be reduced through investigation (e.g., the mean failure time for a batch of light bulbs). Variability is inherent in many systems (e.g., the distribution of failure times for a batch of light bulbs) and cannot be reduced. It is often valuable to explicitly separate variability from uncertainty in a model.

With SubModels, this is accomplished by inserting a SubModel (e.g., a dynamic Monte Carlo simulation) within an outer model (e.g., a static Monte Carlo simulation). This is referred to as "nested" Monte Carlo simulation. The inner model simulates the random variability of the system, while the outer model represents the ignorance in key parameters.

In the example above, the outer model would sample a probability distribution that represents the uncertainty in the mean lifetime of a light bulb, and the inner model would simulate the performance of a number of random light bulbs (whose lifetime is sampled from a distribution with the mean specified by the outer model).

Note that for this kind of model, if you were to link an output from the SubModel to the Output Interface, conceptually on the outside of the SubModel the output represents a *distribution of distributions*. That is, because the SubModel is a Monte Carlo model, every time it is called, it produces a distribution for any result linked to the Output Interface. However, because the outer model itself is a Monte Carlo simulation, every element inside it results in a distribution of results. Hence, running the outer model multiple times results in a distribution of distributions for any SubModel output.

GoldSim 11 provides specialized capabilities to display nested Monte Carlo results. This new advanced feature is described in detail in the GoldSim User's Guide, starting on page 911. Alternatively, open the Help file, and on the Index tab enter *Nested Monte Carlo*.

New Features to Facilitate Modeling Aging Chains

In some situations, it is necessary to keep track of the age structure of a stock of material or objects. For example, you may want to track the number of people in each of a number of age groups, the number of people in a company at different experience levels (e.g., new hire, experienced, expert), or the number of trucks of different age groups on the road.

To model such a situation, you cannot use a single Stock (e.g., a Reservoir). Rather, you must disaggregate the total stock into multiple categories (referred to as cohorts). Each cohort "graduates" (gets pushed) to the next cohort over time (and can only move in one direction). However, each cohort may grow and shrink for other reasons (e.g., if you were modeling a population of people, a particular category could grow due to immigration, and shrink due to emigration and death). Note that these rates of growing and shrinking are likely a function of the specific cohort (e.g., death rates).

In the past, you could model these kinds of systems using either a series of Reservoirs or a series of Material Delays. GoldSim 11 provides a new set of features that are particularly appropriate when you would like to instantaneously "graduate" all the members of one cohort to the next cohort at a specific time (e.g., every year), and repeat the process for a large number of cohorts (e.g., 100 1-year age groups).

To facilitate this, GoldSim 11 adds new capabilities for Discrete Change and Integrator elements. These new features are described in detail in the GoldSim User's Guide, starting on page 734. Alternatively, open the Help file, and on the Index tab enter *Modeling aging chains*, and then select the subtopic *using Integrators with discrete pushes*.

MINOR MODIFICATIONS

Improvements in How GoldSim Compresses and Stores Bitmap-Based Images

In previous versions of GoldSim, bitmap-based images degraded every time the image was viewed. Moreover, the file size increased whenever the file was saved! In GoldSim 11, this bug has been corrected. Bitmap-based images are not degraded and are compressed so as to avoid excessive file sizes.

GoldSim 11 compresses bitmap-based images using an algorithm that favors image quality over image size. This allows for acceptable image sizes, while ensuring that the compressed image is of acceptable quality.

Modified Spreadsheet and Text Export from Time History Result Elements

The spreadsheet and text file export from time History Result elements has changed in GoldSim 11 in order to support new time history features (e.g., Reporting Periods, scenarios).

One of the key changes that will be noted for spreadsheet export is that GoldSim 11 automatically remembers the range that was written to the previous time it carried out an export and clears out that range prior to a new export. This is useful, for example, if you change the number of timesteps (or the duration) between exports.

The new Time History Result element export features are described in detail in the GoldSim User's Guide, starting on page 659. Alternatively, open the Help file, and on the Index tab enter *Result elements*, and select the subtopic *exporting to spreadsheets*.

Importing Units for Data from Spreadsheets

When importing data from a spreadsheet into GoldSim using Spreadsheet element, you can now import the units for the imported data from a cell in the spreadsheet (rather than specifying the units in GoldSim when defining the import location).

This new feature is described in detail in the GoldSim User's Guide, starting on page 839. Alternatively, open the Help file, and on the Index tab enter *Spreadsheet elements*, and then select the subtopic *importing data from the spreadsheet*.

Support for .XLSM Files

The .XLSM format is the macro-enabled Excel format (corresponding to the .XLSX format). In previous versions, this was not fully supported in Spreadsheet elements (e.g., it did not appear as an extension option when selecting an existing Excel file). This is now supported in GoldSim 11.

Resetting an Extrema Element

GoldSim 11 provides access to a standard GoldSim Trigger dialog within Extrema elements that resets the element. When an Extrema is reset by a trigger, the Peak/Valley calculation is restarted on the next update (ignoring everything that has occurred previously, as if it was the start of the simulation again).

This feature is described in detail in the GoldSim User's Guide, starting on page 247. Alternatively, open the Help file, and on the Index tab enter *Extrema elements*.

<u>Treatment of Fields with Numerator and Denominator Both</u> <u>Equal to Zero</u>

In some cases, you may enter an expression into an input field such as X/Y where at some point during the simulation, both X and Y are zero. In most instances, when GoldSim encounters zero in the denominator of an expression, it will display a fatal error and stop the simulation. However, in GoldSim 11, if *both* the numerator and the denominator are zero, GoldSim will not display a fatal error; instead, the resulting expression will be set to zero and the simulation will continue. This can be very convenient, as it allows you to avoid adding complex logic to check for such cases.

Note, however, that GoldSim only sets the expression to zero if both the numerator and denominator are zero. To understand this, consider an example where A and C are equal to zero, and B is non-zero. In this case, the following two expressions would be treated differently by GoldSim:

- A*B/C: would return zero, as A*B would be evaluated first (and considered to be the numerator for C.
- A*(B/C) would result in a fatal error, as B/C would be evaluated first, and B is non-zero.

New Features for Referencing Exposed Outputs of Localized Containers

GoldSim 11 provides new features to make it easier to quickly reference exposed outputs of localized Containers within input fields of other elements. These features also make Containers act like objects with a custom output interface.

In particular, once an output is exposed on a Container, in Goldsim 11 it is displayed within an *Exposed Outputs* folder within the Insert Link dialog (accessed by right-clicking in an input field and selecting **Insert Link...** For example, suppose that the variable X was exposed on the Container Y. If you selected **Insert Link...** from an input field outside of the Container, you could select X from the Exposed Outputs folder:



Note that if the exposed output has a defined alias, this is shown first, followed by the element name in parentheses:



Similarly, when typing output names into an input field, GoldSim's link suggestion feature allows you to easily enter the Container name followed by the list of exposed outputs. If you wish to link to an exposed output of a localized Container, you can do so by typing a period after the name of the Container in the input field. The suggestion box will then list any exposed outputs that exist for that Container.

Using Flags When Running Player Files from the Command Line

Several command lines flags were added that can be used when running the GoldSim Player in a batch file. (These were actually added in 10.5 SP1, but were not formally documented).

Parameter	Description	Comments
-r -run	Opens the GoldSim Player file and immediately runs the simulation.	
-X	Closes GoldSim after all other options have been executed.	Note that unless the –sv parameter is also used, the GoldSim Player file will NOT be saved when the application is closed.
-sv [filename] -save [filename]	Saves the Player file using either the optionally provided output filename, or the input filename (if an output file is not provided).	This option can be used to run and save a Player file.

In particular, the following flags can now be used when running the GoldSim Player from the command line:

These flags are described in detail in the GoldSim Dashboard Authoring Module User's Guide, starting on page 100. Alternatively, open the Help file, and on the Index tab enter *Player files,* and then select the subtopic *running from command line*.

Modifications to Array Functions

Several array functions have been modified in GoldSim 11. In particular, an optional second argument has been added to sumv, sumc, sumr, prodv, prodc, and prodr:

Function	Description	Result
sumv(VC, X)	Sums the items of the vector. If the arguments are conditions, ORs the conditions. X is optional. If positive, the calculation is applied to the first X terms; if negative, the calculation applies to the last X terms. X is ignored if 0 or greater than array size.	scalar
prodv(VC, X)	Computes the product of the items of the vector. If the arguments are conditions, ANDs the conditions. X is optional. If positive, the calculation is applied to the first X terms; if negative, the calculation applies to the last X terms. X is ignored if 0 or greater than array size.	scalar

Function	Description	Result
sumr(MC, X)	Sums the items across each row of the matrix. If the arguments are conditions, ORs the conditions. X is optional. If positive, the calculation is applied to the first X terms; if negative, the calculation applies to the last X terms. X is ignored if 0 or greater than array size.	vector[A], where A is the set of array labels for the rows in the original matrix
prodr(MC, X)	Computes the product of the items across each row of the matrix. If arguments are conditions, ANDs the conditions. X is optional. If positive, the calculation is applied to the first X terms; if negative, the calculation applies to the last X terms. X is ignored if 0 or greater than array size.	vector[A], where A is the set of array labels for the rows in the original matrix
sumc(MC, X)	Sums the items down each column of the matrix. If the arguments are conditions, ORs the conditions. X is optional. If positive, the calculation is applied to the first X terms; if negative, the calculation applies to the last X terms. X is ignored if 0 or greater than array size.	vector[B], where B is the set of array labels for the columns in the original matrix
prodc(MC, X)	Computes the product of the items down each column of the matrix. If the arguments are conditions, ANDs the conditions. X is optional. If positive, the calculation is applied to the first X terms; if negative, the calculation applies to the last X terms. X is ignored if 0 or greater than array size.	vector[B], where B is the set of array labels for the columns in the original matrix

VC: Vector. Can be value or condition.

MC, *MC1*, *MC2*: *Matrix*. *Can be value or condition*. *X*: *Scalar value*.

Support for Multiplying and Dividing the Rows and Columns of a Matrix by a Vector

GoldSim 11 provides the following new operations for manipulating matrices: Matrix*Vector, Vector*Matrix, and Matrix/Vector. In these cases, the array label set of the Vector must match the array label set of either the rows or columns for the Matrix. If it matches that of the rows, then each row of the matrix is multiplied (or divided) by the corresponding term in the Vector. If it matches that of the columns, then each column of the matrix is multiplied (or divided) by the corresponding term in the Vector. If it matches both, the operation is carried out on the rows. In either case, the result is a Matrix of the same order as the original matrix. Note that the operation described here is quite different from the linear algebra operation of multiplying a Vector by a Matrix (or vice versa), with the result being a Vector. That particular operation can be carried out using the specialized array function "Mult".

Improvements to the Milestone Element

GoldSim 11 includes several improvements to the Milestone element. In particular, a new **Results** tab was added that provides some additional results to make it easier to evaluate when milestones are achieved:

lilestone target date/time: 1	1/ 6/2012	12:00:00 AM
lilestone target elapsed time:	20 day	
Result Analysis		
Probability milestone achieve	d:	100.0%
Probability milestone achieve	d by target time	86.4%
Mean lag (for late achieveme	nt):	9.87 day
Show probability distribution:	Elapsed Tin	ne

Pressing the buttons at the bottom of the page displays the probability distribution of when the Milestone was achieved (in terms of calendar time or elapsed time, respectively) *considering only those realizations in which it was achieved*. This is important, since the default result (obtained, for example, by selecting the Etime output and viewing the **Final Value | Distribution Result...**) includes *all* realizations (including those that were not achieved, for which the result is assumed to be Etime = 0).

The upper portion of the Milestone's **Result** tab shows additional results of interest. Two of these results are defined with respect to a defined *target time* that you define (in terms of either Elapsed Time or Date). The target time defaults to the end of the simulation.

The results displayed directly below the target time are then defined as follows:

Probability milestone achieved: This is the probability that the Milestone is achieved during the simulation. Hence, this represents the fraction of realizations that are included in the **Elapsed Time...** or **Date...** plots accessed via the buttons at the bottom of the page.

Probability milestone achieved by target time: This is the probability that the Milestone is achieved by the specified target time.

Mean lag (for late achievement): For those realizations in which the Milestone is achieved after the target time (but before the end of the simulation), this is the mean time after the target time that the Milestone was achieved.

The Milestone element is described in detail in the GoldSim User's Guide, starting on page 361. Alternatively, open the Help file, and on the Index tab enter *Milestone elements*.

Changes to Time Shifting Options for Time Series Elements

GoldSim 11 provides an additional option for shifting Time Series data. In many cases, you will want to use historic data to populate a Time Series. Having done so, you then want to apply that historic data to your simulation (which likely looks forward in time). In order to do this, therefore, you need to time shift the data so that the historic data is applied in an appropriate and consistent manner.

To support this, GoldSim has provided an option to time shift a data series in two different ways: 1) to randomly select a starting point from historic data; or 2) to shift a set of historic data to match the simulation times. GoldSim 11 expands upon the second option so that you can specifically state the year of the data that you wish to begin your simulation from. That is, this option shifts the time series data forward (or backward) by a multiple of a year such that the simulation begins by using data from the specified **Data year to start in**:

Time Shifting Options		ОК
O Use random starting point		Cancel
Data periodicity;	none 👻	Help
Shift data years to align w	vith simulation dates:	
Data year to start in:	1997	
Time Series data will be s from the specified year.	hifted so the simulation begins with data	
Note: if the end of the data of	et is reached as a result of time shifting	

For example, if the actual data set started on 31 July1989 and ended on 31 December 2011, the Simulation Start Date was 1 January 2013, and the **Data year to start in** was entered as 1990, GoldSim would treat the data set such that the data point corresponding to 1 January 1990 would be used for 1 January 2013 (and, assuming daily data was entered, the data point for 2 January 1990 would be used for 2 January 2013, etc.).

This is particularly powerful because the **Data year to start in** can be specified as a function. One way this can be used is to specify a different year for every realization. For example, if **Data year to start in** was defined as "1990 + Realization -1", data would be

used starting from 1990 for the first realization, 1991 for the second realization, and so on.

Time shifting of Time Series is described in detail in the GoldSim User's Guide, starting on page 213. Alternatively, open the Help file, and on the Index tab enter *Time Series elements*, and then select the subtopic *time shifting data*.

Improvements to How Client Machines Access Floating Licenses

Two minor changes were made to how client machines access floating licenses. First, the path to the floating license to which the client machine is connected is now clearly shown in the Licensing dialog:



Secondly, GoldSim 11 now makes it easier to find another license if you can no longer access your floating license. That is, in some circumstances, a floating license that you were previously connected to may no longer be available (e.g., the license folder can no longer be accessed). In this situation, GoldSim will automatically provide you with an opportunity to disconnect from the license so that you can either register a standalone license, or connect to a different floating license (in previous versions, this situation required a Registry edit).

Managing User-Defined Units

In previous versions of GoldSim, it was somewhat difficult to manage user-defined units and control which units were stored with model files and which were stored on your computer. GoldSim 11 has improved management of these units. In particular, there are now two kinds of user-defined units in GoldSim:

• Units that are stored in your system units file, and hence are available to any model that you edit on your computer. The system units file (units.dat) is located in an application data subfolder on your computer (the location of this file differs depending on the operating system). That is, these units are stored separately on your computer (i.e., not just in the model file itself).

• Units that are only stored within the model file itself, and are not available to any other models that you edit on your computer.

The manner in which these user-defined units are defined and managed is described in detail in the GoldSim User's Guide, starting on page 401. Alternatively, open the Help file, and on the Index tab enter *Units*, and then select the subtopic *managing user-defined*.

Opening an Excel File and Selecting a Specified Sheet and/or Range via a Hyperlink Object

Hyperlink objects in GoldSim allow you to open another document from GoldSim (by selecting the "Web or local document" option. Doing so launches the application associated with the specified file and opens the file in the application. For example, if you specified a Microsoft Excel file (*.xls, *.xlsx or *.xlsm), GoldSim would launch Excel and open the file.

When opening a spreadsheet file, in some cases you may want to immediately jump to a specified location (sheet and cell range) when opening the file. GoldSim 11 now provides specialized syntax to support this. The syntax is as follows:

Filename#sheetname!cellrange (e.g., Book1.xlsx#Sheet1!a1:c3)

This new capability is described in detail in the GoldSim User's Guide, starting on page 695. Alternatively, open the Help file, and on the Index tab enter *Hyperlinks*, and then select the subtopic *specifying addresses*.

Command Line Option for Global Database Download

GoldSim 11 provides a new command line argument (-getdb) that executes a global database download for all elements linked to databases. You can also supply an optional date. The date is only required for Yucca Mountain databases. The date must be specified in local format and be enclosed in quotes (e.g., "7 March 2013").

Reading in an Element Note from a Database

GoldSim 11 provides the ability to read in an element Note from a database. The database must be a Yucca Mountain database. The details of the database format are described in detail in Appendix E of the GoldSim User's Guide.

Recording CPU Times for Elements

There are several kinds of elements in GoldSim that can often consume a large amount of CPU time. These are elements in which you may have custom programmed a computationally-intensive calculation. The elements that support this are the Script, Spreadsheet, External (for linking to DLLs) and the External Pathway (for linking to Contaminant Transport DLLs).

In order to support use of these kinds of "custom elements", GoldSim 11 allows you to log the CPU time that they use. In particular, these elements have a checkbox titled **Record CPU times in the run log**. If you check this box, if the element uses more than 1 CPU seconds, a message will be written to the GoldSim run log identifying the element's name, type (i.e., Script), the number of times it was updated, and the total CPU time used.

CHANGES TO THE CONTAMINANT TRANSPORT MODULE

Inclusion of ICRP Radionuclide Database in GoldSim

GoldSim 11 provides an option (within the RT Module) to utilize a large built-in database of radionuclide decay data (species, decay rates and daughter products). In particular, the feature allows you to view over 1200 radionuclides (along with their corresponding stable elements) and to selectively include some of these radionuclides in your simulation. The data is based on the International Commission for Radiation Protection (ICRP)'s Publication 107.

GoldSim provides a variety of tools for viewing and selecting which of the species in the database are to be included in the simulation:

Description: pecify decay: Half-lives pisplay: Modeled species + Daughters Min half-life to show: 1 yr uto-include ICRP daughters with half-lives >= 1 yr and <= 1e+012 yr Species List Number of Modeled Species : 17 Include Row # ID Weight Half-Life I R Modeled daughters (skipped intermediates) 3 Pa-231 231.036 g/mol 32760 yr Ac-227 Ac-227 Pb 207.2 g/mol 201 B B B B 4 Pb-210 209.984 g/mol 22.2 yr X (Bi-210, Po-210, Pb, Hg-206, Ti-206) X 5 Pu-239 239.052 g/mol 24110 yr X U-235 (U-235m)	
pecify decay: Half-lives	
pecify decay: Half-lives ✓ Species set ordering: Aphabetical ✓ isplay: Modeled species + Daughters ✓ Min half-life to show: 1 yr I I yr uto-include ICRP daughters with half-lives >= 1 yr and <=	
isplay: Modeled species + Daughters Min half-life to show: 1 yr uto-include ICRP daughters with half-lives >= 1 yr and <= 1e+012 yr Species List Number of Modeled Species : 17 Include Row # ID Weight Half-Life I R Modeled daughters (skipped intermediates) 3 Pa-231 231.036 g/mol 32760 yr X Ac-227 Pb 207.2 g/mol Image: Contract of the system of the sys	
Modeled species + Daughters Mill Hild to Show. If yr and <= 1e+012 yr Species List Number of Modeled Species : 17 Include Row # ID Weight Half-Life I R Modeled daughters (skipped intermediates) 3 Pa-231 231.036 g/mol 32760 yr X Ac-227 Pb 207.2 g/mol Image: Comparison of the system of the	
Include ICRP daughters with half-lives >= 1 yr and <= 1e+012 yr Species List Number of Modeled Species : 17 Include Row # ID Weight Half-Life I R Modeled daughters (skipped intermediates) 3 Pa-231 231.036 g/mol 32760 yr X Ac-227 Pb 207.2 g/mol X X Ac-227 4 Pb-210 209.984 g/mol 22.2 yr X (Bi-210, Po-210, Pb, Hg-206, Ti-206) X 5 Pu-239 239.052 g/mol 24110 yr X U-235 (U-235m)	
Species List Number of Modeled Species : 17 Include Row # ID Weight Half-Life I R Modeled daughters (skipped intermediates) 3 Pa-231 231.036 g/mol 32760 yr X Ac-227 Pb 207.2 g/mol X Image: Contract of the second	
Species List Number of Modeled Species : 17 Include Row # ID Weight Half-Life I R Modeled daughters (skipped intermediates) 3 Pa-231 231.036 g/mol 32760 yr X Ac-227 Pb 207.2 g/mol X X Ac-227 4 Pb-210 209.984 g/mol 22.2 yr X X (Bi-210, Po-210, Pb, Hg-206, Ti-206) X 5 Pu-239 239.052 g/mol 24110 yr X Y-235 (U-235m)	
Number of Modeled Species : 17 Include Row # ID Weight Half-Life I R Modeled daughters (skipped intermediates) 3 Pa-231 231.036 g/mol 32760 yr X X Ac-227 Pb 207.2 g/mol X X C Ac-227 4 Pb-210 209.984 g/mol 22.2 yr X X (Bi-210, Po-210, Pb, Hg-206, Ti-206) X 5 Pu-239 239.052 g/mol 24110 yr X X U-235 (U-235m)	
Include Row # ID Weight Half-Life I R Modeled daughters (skipped intermediates) 3 Pa-231 231.036 g/mol 32760 yr X X Ac-227 Pb 207.2 g/mol X X Ac-227 4 Pb-210 209.984 g/mol 22.2 yr X X (Bi-210, Po-210, Pb, Hg-206, Ti-206) 5 Pu-239 239.052 g/mol 24110 yr X X U-235 (U-235m)	
3 Pa-231 231.036 g/mol 32760 yr X Ac-227 Pb 207.2 g/mol X X Ac-227 4 Pb-210 209.984 g/mol 22.2 yr X X (Bi-210, Po-210, Pb, Hg-206, Ti-206) 5 Pu-239 239.052 g/mol 24110 yr X X U-235 (U-235m)	
Pb 207.2 g/mol Image: Constraint of the state of the	
4 Pb-210 209.984 g/mol 22.2 yr ⊠ ⊠ (Bi-210, Po-210, Pb, Hg-206, Ti-206) ☑ 5 Pu-239 239.052 g/mol 24110 yr ☑ ☑ U-235 (U-235m)	
S Pu-239 239.052 g/mol 24110 yr 🛛 🖾 U-235 (U-235m)	
G Pu-240 240.054 g/mol 6564 yr X □ U-236 U-236 U	
7 Pu-242 242.059 g/mol 3.75e+005 yr 🛛 🖾 U-238	
8 Ra-226 226.025 g/mol 1600 yr 🛛 🖾 Pb-210 (Rn-222, At-218, Po-218, Rn-218, Pb-214	14, Bi-214, Po-2
9 Ra-228 228.031 g/mol 5.75 yr 🛛 🖾 Th-228 (Ac-228)	F
10 Th-228 228.029 g/mol 1.9116 yr 🛛 🖾 (Ra-224, Rn-220, Po-216, Pb-212, Bi-212, Po-21	212, TI-208, Pb)
11 Th-230 230.033 g/mol 75380 yr 🛛 🕅 Ra-226	
12 Th-232 232.038 g/mol 1.405e+010 yr 🛛 🕅 Ra-228	
13 U-234 234.041 g/mol 2.455e+005 yr 🛛 🔿 Th-230	
X 14 U-235 235.044 g/mol 7.04e+008 yr [X] [X] Pa-231 (In-231)	
X 14 0-235 235.044 g/m01 7.04e+008 yr X X Pa-231 (In-231) 15 U-236 236.046 g/mol 2.342e+007 yr X X Th-232	
X 14 0-235 235.044 g/mol 7.04e+008 yr X X Pa-231 (In-231) 15 U-236 236.046 g/mol 2.342e+007 yr X X Th-232 X 16 U-238 238.051 g/mol 4.468e+009 yr X X U-234 (Th-234, Pa-234, Pa-234m)	

This new capability fir utilizing data from the ICRP database is described in detail in the GoldSim Contaminant Transport Module User's Guide, starting on page 45. Alternatively, open the Help file, and on the Index tab enter *Species*, and then select the subtopic *ICRP database*.

Improvements to the Aquifer Pathway Element

GoldSim 11 includes two changes to the Aquifer element:

• You can now specify a Source Term Length for an Aquifer pathway (in the same manner that you can do so for a Pipe pathway). In particular, you can specify the Source Zone Length parallel to the flow direction in the pathway. Hence, if your pathway had a length of 100m, and you specified a Source Zone Length of 40m, mass would enter the pathway uniformly over the first 40m (rather than at the beginning of the pathway). This is useful, for example, if your Aquifer represents an aquifer which is being loaded from above with mass from source which is of such an areal extent (e.g., a landfill) that it extends over a significant portion of the pathway and therefore cannot be treated as a point source. This feature is

described in detail in the GoldSim Contaminant Transport Module User's Guide, starting on page 165. Alternatively, open the Help file, and on the Index tab enter *Source zone length*, and then select the subtopic *Aquifers*.

• You can control the outflux boundary condition for an Aquifer pathway. In particular, you can choose whether or not to include dispersive and diffusive outfluxes to downstream pathway(s). If you choose to do so, a zero concentration in the receiving pathway is assumed, and dispersive and diffusive transport into downstream pathways is allowed (which would be appropriate, for example, if the downstream pathway represents a rapidly-flowing, "clean" pathway). Otherwise, only advective transport out of the pathway is allowed. This feature is described in detail in the GoldSim Contaminant Transport Module User's Guide, starting on page 162. Alternatively, open the Help file, and on the Index tab enter *Boundary condition,* and then select the subtopic *Aquifer pathways*.

Improvements to the How Cell Pathways Manage Changing Media Amounts

In some models, media volumes in Cell pathways may change rapidly over a timestep (e.g., a pond rapidly emptying or filling). Under these circumstances, calculated concentrations can be inaccurate (and often oscillatory). GoldSim 11 includes powerful new algorithms to better handle concentration calculations when media amounts are changing rapidly.

CONVERSION ISSUES WHEN READING AND RUNNING EXISTING MODELS

GoldSim 11 represents a major upgrade from previous versions. As part of this upgrade, a number of features (e.g., timestepping, result display) have been completely redesigned. All GoldSim 10.5 model files can be successfully read by GoldSim 11. For the most part, we have worked very hard to ensure that these model files will be correctly converted, and no user action will need to be taken. However, in some instances, this was simply not possible. In some cases, you may need to make minor modifications to your file after reading it into GoldSim 11 in order for it to run properly. In no case should these changes be extensive or time-consuming. However, it is important that you be made aware of how your model may be modified (either automatically when GoldSim converts it, or manually by you after reading the model in order for it to run). These potential conversion issues are discussed below.

IT IS CRITICAL THAT YOU READ THIS SECTION BEFORE READING OLDER MODELS INTO GOLDSIM 11.

Note that if you are having difficulty converting a model from GoldSim 10.5 to 11, do not hesitate to contact us at support@goldsim.com.

New Timestepping Options

As described on page 4, the Simulation Settings dialog has been completely redesigned in GoldSim 11 to provide more intuitive and powerful tools for defining the timestepping options. As a result, however, in some cases, your old timestep specifications may not be able to be represented in the exact same way as they were in previous versions.

In most cases, the timestep definition from old models will indeed map directly into GoldSim 11. However, if you used Phases extensively to create timesteps in previous versions of GoldSim, the Phase timesteps may not be able to be exactly replicated in GoldSim 11 (Phases were replaced with several new, more powerful features in GoldSim 11).

Also, in previous versions of GoldSim, you could control whether or not "unscheduled timesteps" were inserted (e.g., to better represent events that fell between scheduled timesteps). By default, these were always inserted, but you could turn them off. In GoldSim 11, you can no longer control this. "Unscheduled timesteps" are always inserted when required. Due to other improvements in the timestepping algorithm, however, it is unlikely that this change will be noticeable to most users. Note that a new option in Time History Result elements now allows you to view all timesteps in a plot (scheduled and unscheduled).

To make sure that your timesteps as represented in GoldSim 11 are appropriate, you should read that section of the documentation describing the new timestepping features, view the Simulation Settings in GoldSim once you read in your old model, and if necessary, make any changes. GoldSim 11 provides a powerful feature for viewing all the scheduled timesteps in a model, so it is very easy to quickly understand where the timesteps will occur. In the rare cases where changes are required, doing so should be very easy (minutes).

New Treatment of Run Properties

As described below on page 7, GoldSim 11 includes a complete redesign of the Run Properties (e.g., ETime, DayOf Week, etc.) to provide more intuitive and powerful tools for referencing these important properties. This includes changing the names and definitions of several Run Properties that existed in previous versions.

In most cases, the Run Properties will be converted seamlessly when reading in old models, and you need not be concerned about them. However, because of the manner in which the definition of the Run Properties have changed, *if you are referencing these in an elapsed time model*, it is critical that you understand how GoldSim converts the files. The table below describes how the conversion takes place, and points out how the definitions have changed (old variables are converted to the new variables when an older file is read into GoldSim 11):

Old Variable	Old Behavior	New Variable	New Behavior	Comments
Time	Elapsed time since beginning of simulation (units of time) when running an Elapsed Time model.	ETime	Elapsed time since beginning of simulation (units of time)	
ETime	Elapsed time since beginning of simulation (units of time)	ETime	Elapsed time since beginning of simulation (units of time)	
Second	Cycles from 1 to 60 (dimensionless integer).	ESecond	Elapsed number of seconds (dimensionless integer).	Starts at 0 at beginning of simulation (not 1). Does not cycle.
Minute	Cycles from 1 to 60 (dimensionless).	EMinute	Elapsed number of minutes (dimensionless integer).	Starts at 0 at beginning of simulation (not 1). Does not cycle.
Hour	Cycles from 1 to 24 (dimensionless integer).	EHour	Elapsed number of hours (dimensionless integer).	Starts at 0 at beginning of simulation (not 1). Does not cycle.
Day	Cycles from 1 to 30; actually cycles back to 1 at multiples of 30.4375 days. (dimensionless integer).	EDay	Elapsed number of days (dimensionless integer).	Starts at 0 at beginning of simulation (not 1). Does not cycle.
Month	Cycles from 1 to 12 (dimensionless integer).	EMonth	Elapsed number of average months (dimensionless integer).	Starts at 0 at beginning of simulation (not 1). Does not cycle. Average month assumed to have 30.4375 days.

Old Variable	Old Behavior	New Variable	New Behavior	Comments
Quarter	Cycles from 1 to 4 (dimensionless integer).	EQuarter	Elapsed number of average quarters (dimensionless integer).	Starts at 0 at beginning of simulation (not 1). Does not cycle. Average quarter assumed to have 91.3125 days.
Year	Current year of simulation, starting at 1 (dimensionless integer).	EYear	Elapsed number of average years (dimensionless integer).	Starts at 0 at beginning of simulation (not 1). Does not cycle. Average year assumed to have 365.25 days.
DayofWeek	Cycles from 1 to 7 (dimensionless integer). Starts at 1.	DayofWeek	Cycles from 1 to 7 (dimensionless integer).	Starts on the actual day specified in the Simulation Start Time field.
DayofYear	Cycles from 1 to 365.25 (dimensionless integer). Starts at 1.	DayofYear	Cycles from 1 to 365.25 (dimensionless integer).	Starts on the actual day specified in the Simulation Start Time field.
WeekofYear	Cycles from 1 to 52 (dimensionless integer). Starts at 1.	WeekofYear	Cycles from 1 to 52 (dimensionless integer).	Starts on the actual week specified in the Simulation Start Time field.

Looking at this table, the following points should be noted when reading in an elapsed time model from a previous version into GoldSim 11:

- There are no issues when converting references to Time or ETime. These are both treated identically (as ETime) in GoldSim 11.
- In old models that reference Second, Minute, Hour, Day, Month, Quarter and Year, GoldSim converts these to new variables with the prefix E (e.g., EYear). It is important to understand that the behavior of the old and the new variables differ in two key ways: 1) the old variables cycled (e.g., from 1 to 60), while the new variables do not cycle (they continue to increase); and 2) the old variables started at 1, while the new variables start at 0 (since they represent true elapsed time variables). If this behavior is no longer appropriate, you may need to modify your model. Note, however, that in the majority of elapsed time models, if a variable

such as Day was being used at all, it was likely being used as a trigger (most likely, within an OnChanged trigger), and in these cases the new variables will produce the same behavior. If, however, you were using a property such as Month as an input to a Lookup Table in an elapsed time model, this will be converted to EMonth in GoldSim 11, and will no longer work as you originally designed (in this case, because it is zero based and does not cycle after reaching 12). You will need to make changes to your model to correct this.

• When reading an old model that references DayofWeek, DayofYear and WeekofYear, GoldSim does not carry out any conversion. However, these three variables are treated differently in elapsed time models in GoldSim 11 than they were treated in earlier versions. In particular, in older versions, these variables always started at 1. In GoldSim 11 (even in elapsed time models), GoldSim uses the actual Start Time to determine the appropriate initial value for these variables.

In general, if the old model was calendar based (date-time) model, there should be no conversion issues.

Finally, it is possible that in some cases, GoldSim will not convert an expression that references a Run Property at all. This could happen if the expression that is being evaluated is invalid (or has become invalid due to a previous conversion when GoldSim is reading the old model). It could also happen if an expression contained multiple Run Properties that needed to be converted. In this case, after reading in an old model, you may find that several elements have invalid entries. Of course, GoldSim will alert you to this, and fixing them (by replacing with the name of the appropriate updated Run Property) should be quite simple.

One potentially high-impact example of this would be an expression (in your old elapsed time model) that referenced Time. In older versions of GoldSim, Time is interpreted as ETime in an elapsed time model. By default, GoldSim 11 converts Time to ETime when reading in an old elapsed time model. However, if for some reason the expression became invalid when it was opened in GoldSim 11, Time would remain in the expression. Hence, after reading in an old elapsed time model (and correcting all expressions so the entire model is valid), you should do an Affects search (from the browser) on Time to see if it is referenced anywhere in the model. If it is, you should change it to Etime.

Use of Keywords in Charts

A number of the keywords have changed in GoldSim 11. New keywords are now available, and more importantly, a number of old keywords are no longer supported. In some cases, old keywords are automatically converted to corresponding new keywords. In other cases, however, old keywords that are no longer supported may not be converted. If a keyword is no longer supported and is not converted, it will simply appear as it was entered (e.g., %en%).

In either case, you should view any chart in which you are using keywords to ensure that the chart is still labeled appropriately. In some cases, it is possible that you may need to edit the Style to account for the changes in keywords.

The keywords supported in GoldSim 11 are is described in detail in the GoldSim User's Guide, starting on page 657. Alternatively, open the Help file, and on the Index tab enter *Keywords in chart styles*.

Data Styles in Charts

In previous versions, all of the properties of a chart could be specified in the Chart Style dialog. In GoldSim 11, unlike the other attributes of a chart (e.g., axes labels, headers, legends), the manner in which the data itself is shown in the chart is not a property of the chart (and hence is not edited in the Chart Style dialog), but is determined by the type of result and the context in which it is being plotted (e.g., is the result an array, are scenarios being viewed, have classification categories been defined).

As such, Data Styles can be specified in multiple places within GoldSim and the style that is used for a particular chart is a function of the context for the display.

What this means for conversion is that when you read in files containing Result elements, the data style (e.g., line style and color) may appear different in GoldSim 11 (since they will revert to the default for that particular type of chart). Hence, you may want to edit the Data Style if you want the charts to look identical to those in previous versions.

Note that GoldSim 11 will import all other chart style settings for Result elements (e.g., axes labels and fonts). As discussed above, however, there may be some old keywords that will need to be modified (e.g., in Headers).

Controlling Data Styles in GoldSim 11 is described in detail in the GoldSim User's Guide, starting on page 650. Alternatively, open the Help file, and on the Index tab enter *Data styles*.

Change in How "Items" Unit Category is Defined

In previous versions of GoldSim, the "item" unit (in the "Items" unit category) had dimensions of 1/A, where A was a Quantity of Matter, defined by the base unit mol. Hence, 1 item was equal to 1.66E-24 mol. In GoldSim 11, the "Items" category (and all the units in it, such as "item" or user-defined units that you have created) are dimensionless.

If you are using a unit in the Items unit category (i.e., the unit "item" and any userdefined units that you have previously created such as fish or widget) will be automatically converted when you read your model into GoldSim 11. That is, the units will be converted from having a dimension of Quantity of Matter to being dimensionless.

In most models, the model will be automatically converted with no issues. There are two problems that can arise, however, that you must be aware of:

1) If you have created a Lookup Table or a Time Series in which the output display units are from the Item category, GoldSim treats the numbers defining the table or

time series as if the original values had units of 1/6.02E23 moles. Hence, if the table contained a value of 1 Item in the old version, in the new version, this will have been converted to 1.66e-024 items in the new version! This is obviously not what you want. To handle these situations, you will need to have two versions of the file open: the old version, and the GoldSim 11 version. You will then need to cut and past the data from the Lookup Tables and Time Series whose output was a function of something in the Item category from the old version of the file to the new version of the file.

2) In rare cases, when the model is imported, some expressions could become invalid. This would happen, for example, if an expression in an old version of GoldSim added "mols" to "items". The old version would have accepted this (since both units in older versions were Quantities of Matter), but GoldSim 11 would report incompatible units (since in GoldSim 11, "items" is dimensionless). For these expressions, you will need to manually edit them in an appropriate manner.

Log Interpolation of Independent Variables for Lookup Tables No Longer Supported

Previous versions of GoldSim allowed you to specify log interpolation for both the independent and dependent variables in a Lookup Table. Starting in GoldSim 11, you can no longer specify log interpolation for independent variables.

This feature was rarely used, and eliminating it facilitated some additional features added to Lookup Tables for this version (discussed below), including doing reverse lookups, and computing the integral and derivative of the lookup table at a particular location.

If you read in a file that included a Lookup Table that used log interpolation on an independent variable, GoldSim automatically converts it to linear interpolation, and alerts you via a warning message.

Note that if it is truly necessary to use log interpolation on an independent variable, you can implement this by entering the logs of the independent variable when defining points in the Lookup Table (and using the log of the independent variable as an argument when referencing the table).

<u>Changes to How SubModels Exchange Probabilistic Information</u> with the Parent Model

A major change was made in GoldSim 11 to how SubModels pass Monte Carlo results through the output interface to the parent model. In previous versions of GoldSim, when Monte Carlo results were exported from a SubModel via the output interface, you were

required to specify a particular statistic (e.g., mean), and that is what was actually made available on the interface.

In GoldSim 11, Monte Carlo results from a SubModel are actually output in the form of a Distribution output. This output is a complex output representing all the distribution information.

The details of how to select items for the output interface of a SubModel are described in detail in the GoldSim User's Guide, starting on page 906. Alternatively, open the Help file, and on the Index tab enter *SubModels*, and then select the subtopic *output interface*.

This new ability is very powerful, because rather than simply exporting a single statistic, you can export an entire distribution, and then subsequently query it for a range of statistics, using GoldSim's new distribution functions (see *New Functions that Operate on Distributions* on page 27 above). Alternatively, you could use the output to define the properties of a Stochastic element in the parent model (using the new Externally-Defined Stochastic type).

However, due to this change, <u>any existing model that contains probabilistic SubModels</u> (i.e., any SubModel for which the Solution Type was "Simulation" and did not have "Deterministic" selected for its Monte Carlo option) will need to be edited in order to run in GoldSim 11. This is because for these models, in previous versions you were required to specify a particular statistic (e.g., mean), and that is what was actually made available on the interface. When these models are imported into GoldSim 11, however, these interface outputs will no longer be single statistical values. Instead, they will be a complex Distribution output. Hence, wherever they are being referenced in the parent model, the input will be invalid. Editing the model is straightforward. Instead of using the old interface output directly, you will need to create an appropriate Expression element in the parent model that uses the new distribution functions provided by GoldSim 11. These functions operate on a Distribution output, and produce a statistic (e.g., mean, SD). You would then use the output of the new Expression in the location where you were previously using the old output that was coming directly from the SubModel output interface.

Similarly, if the Solution Type for the SubModel is "Simulation" and the Monte Carlo options are set to "Probabilistic", only scalar values and conditions (and Time Series definition) outputs can be exposed on the output interface of the SubModel. In previous versions, array values could be selected. If you read in an old model that contains a Probabilistic SubModel with an array on the output interface, GoldSim will display an error and you will need to edit the model. Note, however, that if you read in an old model in which the Solution Type is "Simulation" and the Monte Carlo options are set to "Deterministic" or if the Solution Type is "Optimization", array values and conditions can be selected.

SubModels No Longer Allow You to Define Number of Timesteps on the Input Interface

SubModels allow you to define some of their simulation settings from outside the SubModel (i.e., using values provided by the outer model). In previous versions of GoldSim, one of the simulation settings you could define in this way was the "Number of Timesteps".

This is no longer supported in GoldSim 11 (as the number of timesteps is no longer specified; instead you define the timestep length). If this is encountered when reading in an old model, it is removed, and a note is written to the screen.

Changes to How Time History Result Elements Export to Spreadsheet and Text Files

The spreadsheet and text file export from time History Result elements has changed in GoldSim 11 in order to support new time history features (e.g., Reporting Periods, scenarios).

Hence, if you are actively using this feature to export data, you may need to make some modifications to your procedures or post-processing functions to reflect the new behavior.

The new Time History Result element export features are described in detail in the GoldSim User's Guide, starting on page 659. Alternatively, open the Help file, and on the Index tab enter *Result elements*, and select the subtopic *exporting to spreadsheets* or *exporting to text files*.

Changes to Input Requirements for Event Delays and Timed Events Defined Using Stochastics

In GoldSim 11, Stochastic elements have an additional output (the Distribution definition). This output is a complex output representing all the distribution information.

The Timed Event element has an option to define the Occurrence Type as "Stochastic element time intervals". In previous versions, this required the input for the Occurrence Rate to be the primary output of the Stochastic element. In GoldSim 11, the new Distribution output of the Stochastic must be referenced:

Event Definition		
Occurrence Type:	Stochastic element time intervals	•
Stochastic:	Stochastic1.Distribution	

Similarly, the Event Delay has an option to define the Delay Type as "Stochastic Delay Time Definition". In previous versions, this required the input for the Stochastic to be the

primary output of the Stochastic element. In GoldSim 11, the new Distribution output of the Stochastic must be referenced:

Delay Definitio	n	
Delay Type:	Stochastic Delay Time Definition	•]
Stochastic:	Stochastic 1. Distribution	

GoldSim will automatically convert this input, so it should not be necessary for you to take any action in this regard.

Changes in Advanced Timestep Settings

In GoldSim 10.5, GoldSim offers two settings to control if and how "unscheduled" timesteps can be inserted dynamically into a model:

dvanced Time Settings	×
I Allow events to occur between timesteps	
Minimum timestep length to allow: 0.0 day	
Allow dynamic reduction in timestep length	
Maximum timestep length to allow:	

In GoldSim 11, instead of two independent settings, these two settings are nested. That is, the second setting can only be enabled if the first is enabled:

Minimum time between updates:	0.0 day
Dynamically control time betwee	en undates
V Dynamically control time betwe	en updates

The default for new models is to activate the first setting with a minimum time of zero.

In existing models in which only the second setting was active, GoldSim will automatically activate the first setting (and set the minimum time to zero).

Displaying Results in a Dashboard

In GoldSim 10.5, if you created a Button in a Dashboard that displayed a result (by specifying a Result element), you also needed to specify the Type of result to display (Table or Chart).

In GoldSim 11, you no longer specify the Type. Instead, the Result element remembers the last state of the display (i.e., table or chart), and displays that when the button is pressed.