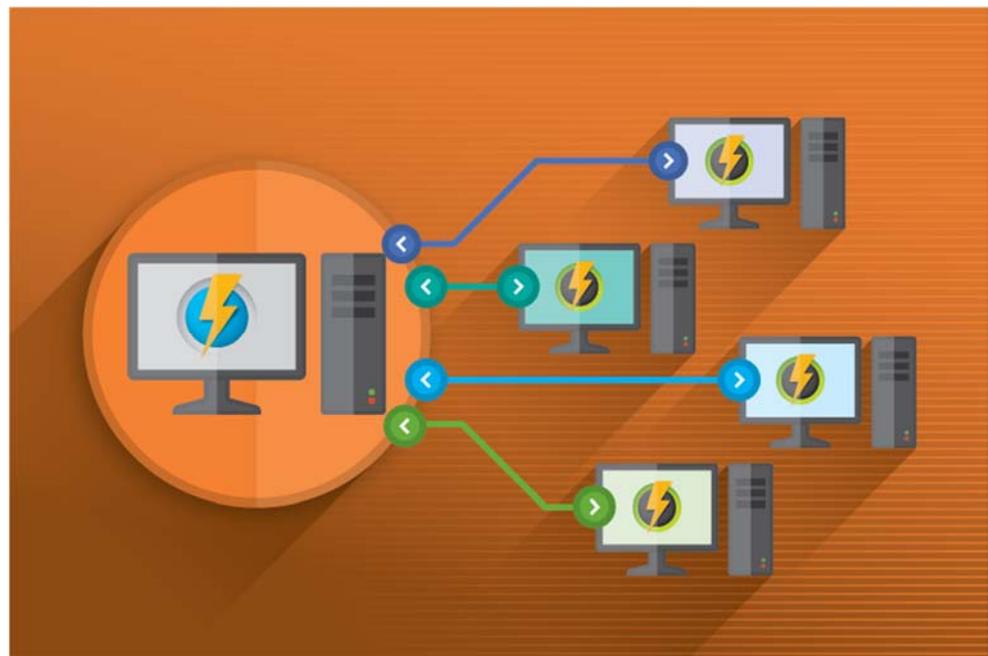


# GoldSim Distributed Processing Module





Copyright GoldSim Technology Group LLC, 2001-2021. All rights reserved.  
GoldSim is a registered trademark of GoldSim Technology Group LLC.

Version 14.0 (October 2021)



GoldSim Technology Group  
255 S. King Street, Suite 800  
Seattle, Washington 98104  
USA

Visit us at our web site: [www.goldsim.com](http://www.goldsim.com)  
Email us at: [software@goldsim.com](mailto:software@goldsim.com)



---

# Contents

<b>Chapter 1: Introduction</b>	<b>1</b>
Chapter Overview .....	1
In this Chapter.....	1
<b>What is the Distributed Processing Module? .....</b>	<b>2</b>
<b>Which Version (DP or DP Plus) Do I Have? .....</b>	<b>2</b>
<b>Conventions Used in this Manual .....</b>	<b>3</b>
<b>Using Help.....</b>	<b>4</b>
<b>Technical Support, User Resources and Software Upgrades .....</b>	<b>4</b>
GoldSim Maintenance Program .....	4
Getting Technical Support .....	4
Other GoldSim Resources.....	5
<b>Chapter 2: Using the Distributed Processing Module</b>	<b>7</b>
Chapter Overview .....	7
In this Chapter.....	7
<b>Overview of the Distributed Processing Module .....</b>	<b>8</b>
<b>Carrying out a Distributed Processing Simulation.....</b>	<b>8</b>
Running GoldSim in Client Mode .....	8
Connecting the Controller to the Clients.....	12
Running a Distributed Processing Simulation.....	19
<b>Limitations of the Distributed Processing Module .....</b>	<b>26</b>
<b>Glossary of Terms</b>	<b>29</b>
<b>Index</b>	<b>31</b>



---

# Chapter 1: Introduction

Two heads are better than one.

Anonymous

## Chapter Overview

GoldSim is a user-friendly, highly graphical, object-oriented program for carrying out dynamic, probabilistic simulations to support management and decision-making in engineering, science and business.

The GoldSim Distributed Processing Module is a program extension to GoldSim which allows you to use multiple computers connected over a network to share the computational burden of a Monte Carlo simulation.

Each of the GoldSim modules has a separate User's Guide describing its capabilities and features. This short document provides a complete description of the features and use of the GoldSim Distributed Processing Module.



**Note:** This document only describes the Distributed Processing Module, and assumes that you are familiar with the basic capabilities of GoldSim (described in the **GoldSim User's Guide**).

---

### In this Chapter

This introductory chapter discusses the following topics:

- What is the Distributed Processing Module?
- Which Version (DP or DP Plus) Do I Have?
- Conventions Used in this Document
- Using Help
- Getting Technical Support

## What is the Distributed Processing Module?

GoldSim was designed to provide a highly efficient simulation framework for use on a single Windows personal computer. For most applications, running your GoldSim application on a single machine will be sufficient. For highly complex models, however, running Monte Carlo simulations on a single machine could become impractical. For example, if each realization took one minute to complete, then 1000 realizations would take over 16 hours.

The Distributed Processing Module allows you to overcome this problem by combining the power of multiple cores and/or computers to carry out a simulation involving multiple realizations.

This is accomplished by having a **Controller** GoldSim executable connect to an unlimited number of **Client** GoldSim executables. These Client processes can be on a single computer or on computers connected over a network. The Controller assigns and sends individual realizations to the Client processes. Each Client carries out its assigned realization, sends the results to the Controller, and then makes itself available to work on another realization. Once all of the realizations have been completed, the Controller process combines all of the results into a single GoldSim file (and enters Result Mode) just as if the entire simulation had been carried out on the Controller's machine.

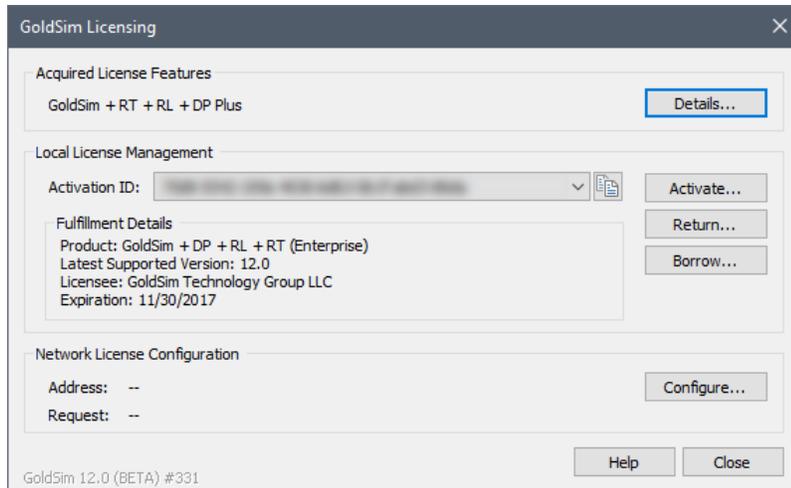
Although the Controller and Client processes will typically be connected via a Local Area Network (LAN), they can also communicate via a Wide Area Network (WAN) or even the Internet. In general, the speed of the simulation is proportional to the number of available Client processes.

There are two versions of the Distributed Processing Module. **GoldSim DP** comes bundled with all versions of GoldSim, and limits the number of Client processes to four. **GoldSim DP Plus** allows an unlimited number of Client processes, and must be licensed separately.

## Which Version (DP or DP Plus) Do I Have?

As pointed out above, there are two different versions of the Distributed Processing Module, and your copy of GoldSim will have one or the other. You can determine which version of the Distributed Processing Module is activated on your machine by selecting **Help|Licensing** from the main menu.

At the top of the dialog, the "Acquired License Features" section will list "DP Plus" if you have the DP Plus version. If you don't see "DP Plus", it means you have the standard version of the module that limits the number of Client processes to four.



## Conventions Used in this Manual

The following conventions are used in this manual:

Convention	Description
<b><i>Important Terms</i></b>	New and important terms are presented in <i>bold italics</i> . These terms all appear in the Glossary of Terms at the end of the document.
<b>File   Open...</b>	Menus and menu selections are separated by a vertical bar. <b>File   Open...</b> means "Access the File menu and choose Open"
CTRL+C	Key combinations are shown using a "+" sign.. CTRL+C means press the Control and C keys simultaneously.
	<b>Warning:</b> This means watch out! Warnings typically alert you to potential pitfalls and problems that may occur if you perform (or fail to perform) a certain action.
	<b>Note:</b> Notes highlight important information about a particular concept, topic or procedure, such as limitations on how a particular feature can be used, or alternative ways of carrying out an action.

In describing various mouse actions, the following conventions are used:

Mouse Action	Definition
Click	Press and release the left mouse button once.
Double-click	Press and release the left mouse button twice in rapid succession.
Right-click	Press and release the right mouse button once.
Drag	Press the left mouse button, and while keeping it depressed, move the cursor to another location, then release the button.

## Using Help

GoldSim has an extensive in-product help facility, which can be used to supplement this manual. The Help system can be accessed by selecting **Help | Help Topics** from the main GoldSim menu or the Help button (the question mark) on the standard toolbar. Pressing the **Help** button within any of the dialogs also provides access to Help (in a context-sensitive manner). All of the information in this document is accessible via GoldSim Help.

## Technical Support, User Resources and Software Upgrades

The GoldSim Technology Group is dedicated to providing complete solutions for our customers. We pride ourselves in providing prompt and extensive support and resources to our users, and are committed to ensuring that each installation of our software is successful and adds value to the customer.

### GoldSim Maintenance Program

When you purchase GoldSim software, you receive one year of Software Maintenance, entitling you to the following:

- Free software upgrades so that you always have the latest version of the GoldSim software.
- Basic Technical Support via email and phone. Basic support covers installation and licensing questions, as well as questions about GoldSim's features and capabilities.

After the first year, if you wish to continue to have access to new versions and technical support, Software Maintenance can be extended each year with payment of an annual fee.

Details regarding the GoldSim Maintenance Program can be found at [www.goldsim.com/Web/Products/BuyGoldSim/Pricing/MaintenanceProgram/](http://www.goldsim.com/Web/Products/BuyGoldSim/Pricing/MaintenanceProgram/).

### Getting Technical Support

Users with active Software Maintenance can submit questions directly to the GoldSim support team. Evaluation users are also welcome to contact us with questions on GoldSim functionality. The **GoldSim Help Center** (<https://goldsim.zendesk.com>) is the primary portal for technical support. You can submit your questions directly from the Help Center. If you register and log in through the Help Center, you will be able to check the status and view a history of all of your support requests.

The Help Center also includes:

- The **GoldSim Forum**, where you can post questions to the GoldSim community, or just browse existing messages;
- Articles on licensing questions and modeling tips; and
- An archive of past webinars (which demonstrate GoldSim features and capabilities).

Free Basic Technical Support does not include consulting, model troubleshooting or detailed assistance with applying GoldSim to a particular problem. Assistance of this nature is defined as Advanced Technical Support. Users may purchase Advanced Technical Support in pre-paid 10 hour blocks.

Details regarding Advanced Technical Support can be found at [www.goldsim.com/Web/Resources/TechnicalSupport/](http://www.goldsim.com/Web/Resources/TechnicalSupport/).

## Other GoldSim Resources

In addition to the GoldSim Help Center, additional resources are also available. These three resources can be accessed directly from the GoldSim website ([www.goldsim.com](http://www.goldsim.com)):

- A free **Online Training Course** that will provide you with a thorough understanding of the key concepts on which GoldSim is based and all of the fundamentals required to build complex models of nearly any kind of system.
- The **GoldSim Model Library**, which contains a collection of example models to allow you to see how specific features of GoldSim can be used and/or how GoldSim can be used for specific applications.
- The **GoldSim Blog**, which provides an informal mechanism for GoldSim staff to share their knowledge, point out some of the more advanced (and perhaps overlooked) GoldSim features, share and discuss common mistakes we see in GoldSim applications, discuss interesting applications, and keep you abreast of our plans for further GoldSim developments.

You can stay up to date on the latest GoldSim news through these resources:

- The GoldSim LinkedIn Group, which is primarily used for announcements (e.g., new versions, interesting applications). You can join the Group here: [www.linkedin.com/groups/1798413](http://www.linkedin.com/groups/1798413)
- Periodic email newsletters are sent two to three times per year. To be added to the newsletter list, contact us via the GoldSim Help Center (<https://goldsim.zendesk.com>).



**Note:** When you purchase GoldSim, you are entitled to a free one hour, live web-based training session in which one of our analysts provides an interactive training session via the Internet and telephone. You are strongly encouraged to take advantage of this free training.

---



---

# Chapter 2: Using the Distributed Processing Module

R2-D2, you know better than to trust a strange computer.

C-3PO

## Chapter Overview

This chapter provides the details about using the GoldSim Distributed Processing Module. After first providing an overview of how a distributed processing simulation is carried out, step-by-step procedures for carrying out such a simulation are described. The chapter closes with a discussion on the limitations of the Distributing Processing Module.

### In this Chapter

This chapter discusses the following topics:

- Overview of the Distributed Processing Module
- Carrying Out a Distributed Processing Simulation
- Limitations of the Distributed Processing Module

## Overview of the Distributed Processing Module

The Distributed Processing Module uses multiple copies of GoldSim running on multiple machines (and/or multiple processors or cores within a single machine) to share the computational burden of a Monte Carlo simulation.

A distributed processing simulation is controlled by a single *Controller* GoldSim executable. The Controller executable is a version of GoldSim running on one machine which has the Distributed Processing Module enabled. Through a dialog accessed via a menu item, the Controller can create connections with multiple *Client* GoldSim processes.

The Controller assigns and sends individual realizations to the Client processes. Each Client process carries out its assigned realization, sends the results to the Controller, and then makes itself available to work on another realization. Once all of the realizations have been completed, the Controller combines all of the results into a single GoldSim file (and enters Result Mode) just as if the entire simulation had been carried out on the Controller's machine.

The GoldSim model file (and other required files such as user-created DLLs or linked spreadsheets) need not be present on machines running Client processes. The Controller automatically transfers any necessary files to the Client machines.

When running as a Client process, GoldSim is run in a special mode. Rather than launching GoldSim in the traditional manner on the machine running the Client, a GoldSim Client is launched from the command line (this can be done using a Windows shortcut or the Run option on the Windows Start menu). When running in Client mode, you cannot view or edit a GoldSim model. The only function that a Client serves is to receive, process, and send back realizations sent to it by a Controller.

In order for the Controller process to access the Client processes, they must be local to the computer or accessible over a network.

## Carrying out a Distributed Processing Simulation

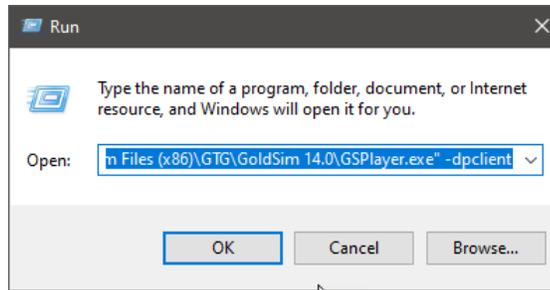
The steps required to carry out a distributed processing simulation are as follows:

1. Launch one or more GoldSim processes on the machine(s) that you wish to utilize.
2. Open the GoldSim model file and connect the Controller to the Client processes.
3. Run the simulation.

Each of these three steps is discussed in detail below.

### Running GoldSim in Client Mode

In order to start a GoldSim Client process on a machine, GoldSim (or the GoldSim Player) must be installed. The GoldSim Player is automatically installed when GoldSim is installed, and is also available as a free download. To launch the GoldSim Player in Client mode, you must run the Player from the command line using the “-dpclient” parameter. Typically, this would be done by accessing the Windows Run option, which displays the following dialog:

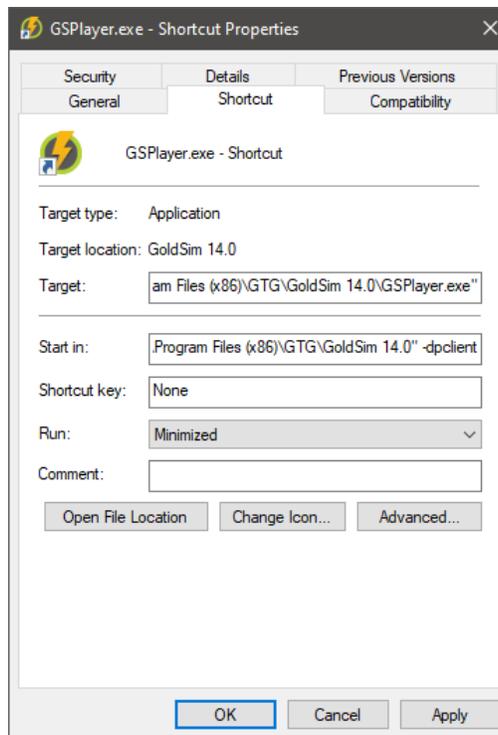


You must enter the full path to the GoldSim Player executable (which may, of course, be different from that shown above), followed by the “-dpclient” command line parameter.



**Note:** In addition to using the GoldSim Player as a Client, any GoldSim executable (even if it is unregistered) can also be accessed as a Client. In this case, of course, the command line to run the Client would reference the GoldSim executable (GoldSim.exe) rather than the Player executable.

You could accomplish the same thing by creating and running a shortcut to the GoldSim Player executable with the appropriate command line parameter:



*In this particular case, the GoldSim Player would be launched in Client mode and immediately minimized.*

If you placed this shortcut in the StartUp directory of the machine, the GoldSim Player would be loaded in Client mode whenever the machine was booted.



**Note:** When creating the shortcut, the pathname to the Player file must be enclosed in quotes, with the `-dpclient` option (to start the Player in Client mode) following the pathname. For example, if the path to the GoldSim Player was `C:\Program Files (x86)\GTG\GoldSim 14.0\GSPlayer.exe`, the Target field of the shortcut dialog should be `"C:\Program Files (x86)\GTG\GoldSim 14.0\GSPlayer.exe" -dpclient`.

---

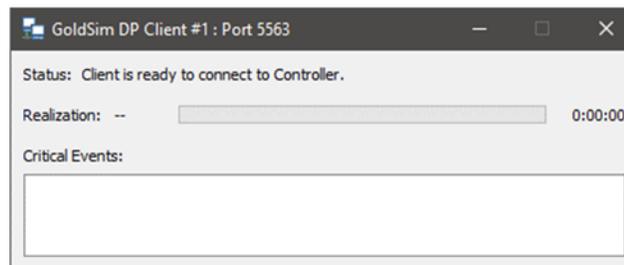
A Client process requires approximately 6Mb of memory when it is loaded prior to being accessed by the Controller process. Hence, for most machines, the act of loading the Client process will not significantly impact anyone who is using that particular machine. Moreover, once a Controller process accesses the Client (and begins to send it realizations to process), the priority of the Client is lower than that of other processes running on the machine. As a result, although an active Client will slow the system down somewhat, the machine can still be used for other applications, with the GoldSim Client process running in the background.



**Note:** You can run multiple Client processes on a multi-core or multi-processor computer to increase performance. For a multi-processor computer, the recommended number of GoldSim processes (Controller and Client) is equal to the number of processors. For a multi-core computer, the recommended number of GoldSim processes (Controller and Client) is equal to the total number of cores minus one. You must launch each process separately (such that each process is present in the Windows Task Bar).

---

If you launch a Client process from the command line, the following dialog will be displayed on the Client machine:



Pressing the **X** in the upper right-hand corner will terminate the Client process (close the GoldSim Client executable) so that it can no longer be accessed by a Controller. A Client can also be terminated remotely by the Controller.

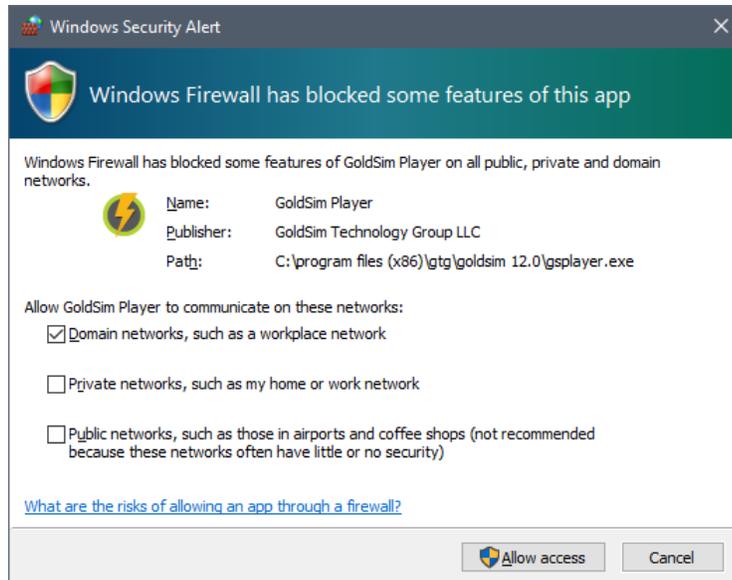
**Read more:** [Terminating Clients from the Controller Process](#) (page 17).

### **Configuring Windows Firewall to Permit Connections to Client Processes**

If a machine has the Windows Firewall enabled, GoldSim or the GoldSim Player must be added to the Windows Firewall's Exception List in order to manually start Client Processes.

The first time that GoldSim or the GoldSim Player is started as a Client process manually, the Windows Firewall will ask if you wish to unblock the program.

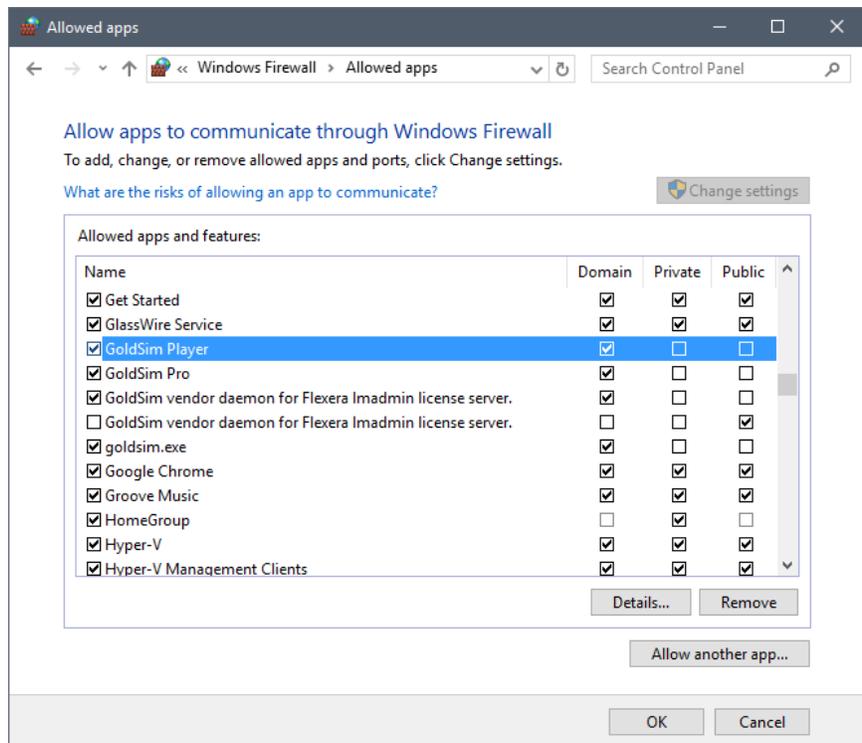
In Windows 10 you will see the following:



For security purposes, you should ensure that only the Domain Networks option is checked. Then click the **Allow Access** button.

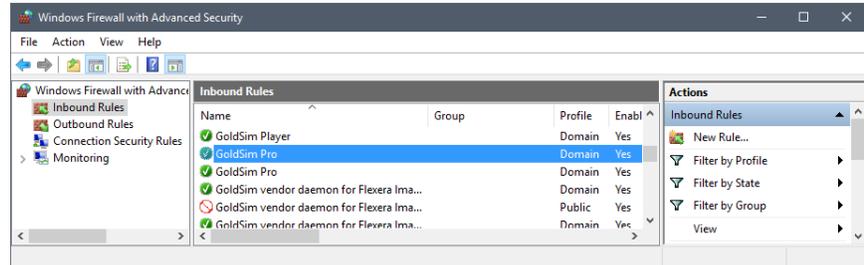
If you were not prompted to unblock the program, it may already have been added to the Windows Firewall Exception List.

You can check that the program has been correctly added by opening the Control Panel, clicking on Windows Firewall and then clicking (in the upper left) “Allow an app or feature through Windows Firewall”.

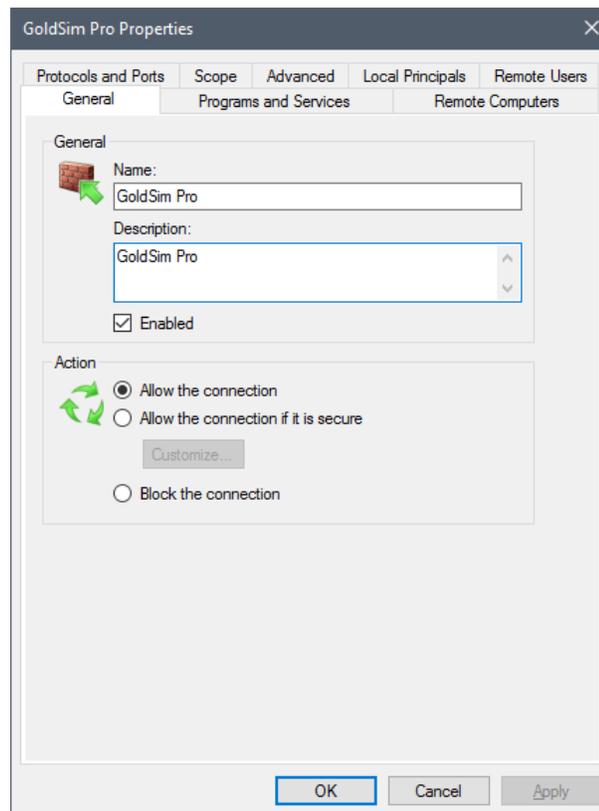


GoldSim or the GoldSim Player will be listed. If the box to the left is checked, the program is already unblocked. If it is unchecked, simply check the box to unblock the program.

You can check that the program has been correctly added by clicking the Start button on the Desktop and then typing Windows Firewall into the Search field. Select *Windows Firewall with Advanced Security* from the Search results (note, to get this to appear in the search results, it may be necessary to type more than *Windows Firewall*). When the program opens, select Inbound Rules on the left-hand side. You'll see the following dialog:

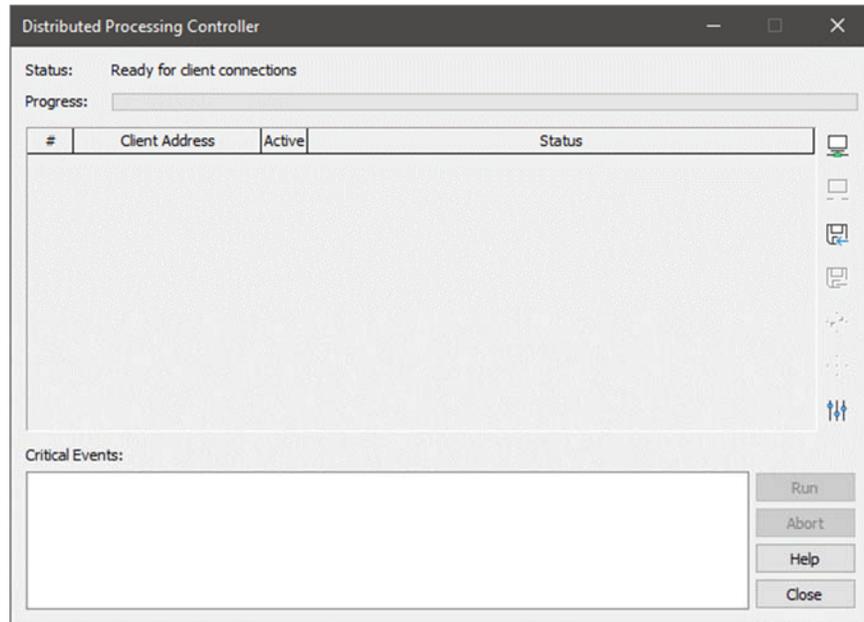


GoldSim Pro and/or the GoldSim Player should show Yes under the Enabled column. If they are not enabled, double click the entry and check the Enabled box on the General tab of the dialog that appears:



## Connecting the Controller to the Clients

To start a distributed processing simulation from a Controller GoldSim executable, start GoldSim, load the model file which you wish to run, and select **Run | Run on Network...** from the main menu. The following dialog will be displayed:



**Note:** The option to **Run | Run on Network...** is only available if multiple realizations have been specified. That is, a distributed processing simulation cannot be carried out on a single realization.

The next step is to specify the names of the machines that will host the Client processes.



**Note:** The list of computer addresses for Client processes is saved to the Registry when you close the DP Controller dialog, so that the addresses you used the last time you ran a distributed processing run will be available the next time the Run on Network dialog is opened.



**Note:** You can assign up to 32 Clients to one computer. The DP clients use ports 5563 through 5594.

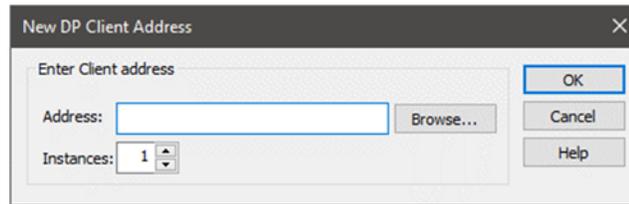
If multiple Client processes will be run on a single PC, the name of that PC should appear as many times as the desired number of Client processes. You can add machine names manually, or by importing a list of Clients that you have previously entered and saved.

**Read more:** [Importing and Exporting Client Lists](#) (page 17).

To add Clients, press the **Add client** button:



This will display the following dialog:



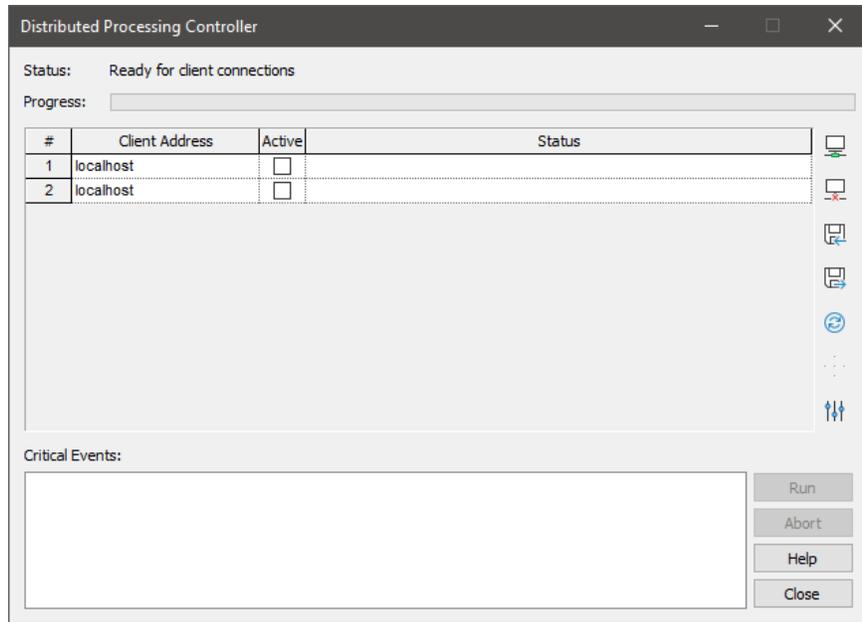
You can use the **Browse...** button to find a computer on your network, or you can enter a network ID or IP address directly. The **Instances** control allows you to create multiple instances of the same computer/address (in order to use multiple processors on that machine). This number must be between 1 and 32, inclusive.



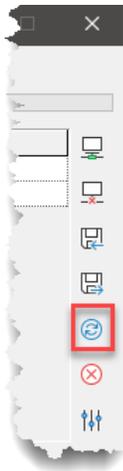
**Note:** If a Client Process is to be run on the local machine, you can simply type *localhost* for the computer name. If you are running multiple Clients on your computer (i.e., using multiple processors) you should specify *localhost* as the client (e.g., if you wanted to use two local processors, you would specify *localhost* as the **Address** and set **Instances** to 2).

---

When you add a Client (and specify the **Instances**) and press the **OK** button, the Client is added to the list. In this example, two local processors have been specified:



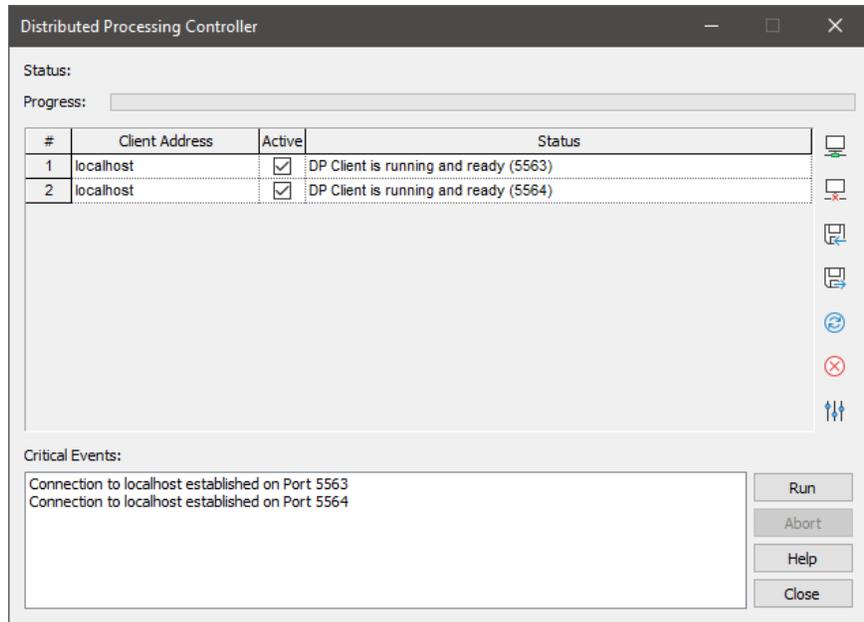
After adding Clients you can activate them by pressing the **Update clients** button:



The Controller then does the following:

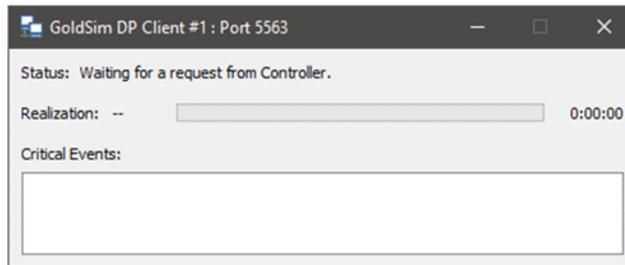
1. Finds the specified computers on the network;
2. Checks to see if a GoldSim Client process is running and available on the computers.

GoldSim then reports the status of the Clients in the dialog:

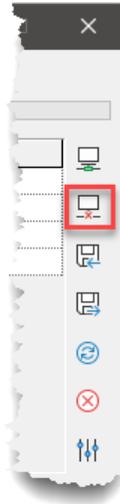


**Note:** Unless you have licensed the Distributed Processing Plus (DP Plus) module, you will be limited to using four Client processes at a time. If you have listed more than four Client processes, when you press the **Update clients** button, GoldSim will only activate four, and the other Client processes will be inactive.

The dialog for Client processes indicates when a connection with a Controller has been made (and that it is waiting for a request):



You can subsequently remove Client process entries from the list in the Controller dialog by selecting an address and pressing the **Remove client** button:

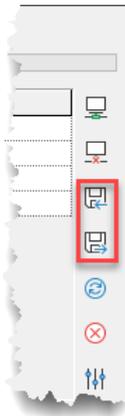


If necessary, existing addresses in the Client process list can also be edited directly.

### ***Importing and Exporting Client Lists***

The list of Client computer addresses is saved to the Registry when you close the Controller dialog, so that the addresses you used the last time you ran a distributed processing run will be available again the next time you wish to carry out a distributed processing run (and you will not have to reenter them).

In some cases, however, you may use different sets of Clients for different runs, or you may want to copy a list of Clients from one computer to another. To facilitate this, GoldSim allows you to export and import lists of Clients using the **Import client list** and **Export client list** buttons:



Once you have entered a list of Clients, pressing the **Export client list** button prompts you to specify a filename for the list of Clients. The file has a .DPL extension. You can subsequently import this list of Clients by pressing the **Import client list** button and specifying a .DPL filename. When you do so, GoldSim will ask if you want to remove any existing Client processes that are already in the dialog and will ask if you want to update the status of all Client processes immediately.

### ***Terminating Clients from the Controller Process***

You can terminate (close) a Client process from the machine on which the Client is running so that it can no longer be accessed by the Controller.

**Read more:** [Running GoldSim in Client Mode](#) (page 8).

You can also terminate Clients from the Controller machine. To terminate all Clients, press the **Terminate all clients** button:



This is typically what you would do when you complete your simulation. You can terminate an individual Client process by clearing the **Active** checkbox. When you do this, you will be presented with a dialog asking if you want to terminate the Client process or simply make it inactive (i.e., stop using it, but keep it open).

**Read more:** [Activating, Deactivating and Terminating Client Processes During a Simulation](#) (page 22).



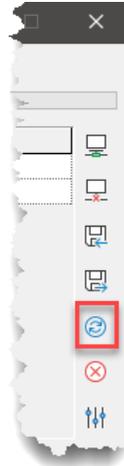
**Warning:** Although the Controller can terminate Client processes, it cannot launch them. A Client process can only be restarted by opening the process locally on the remote machine.

---

### **Updating the Status of Client Processes**

In some situations, the status of a Client could change in such a manner that the Controller would not be aware of the change (until it tried to send realizations to the Client). This could happen, for example, if a Client machine was turned off, or if it was rebooted prior to the beginning of the simulation.

The **Update clients** button provides a mechanism by which the Controller can recheck the status of the Clients prior to starting a simulation (in case some action needs to be taken, for example, if one of the Client machines has been disabled):

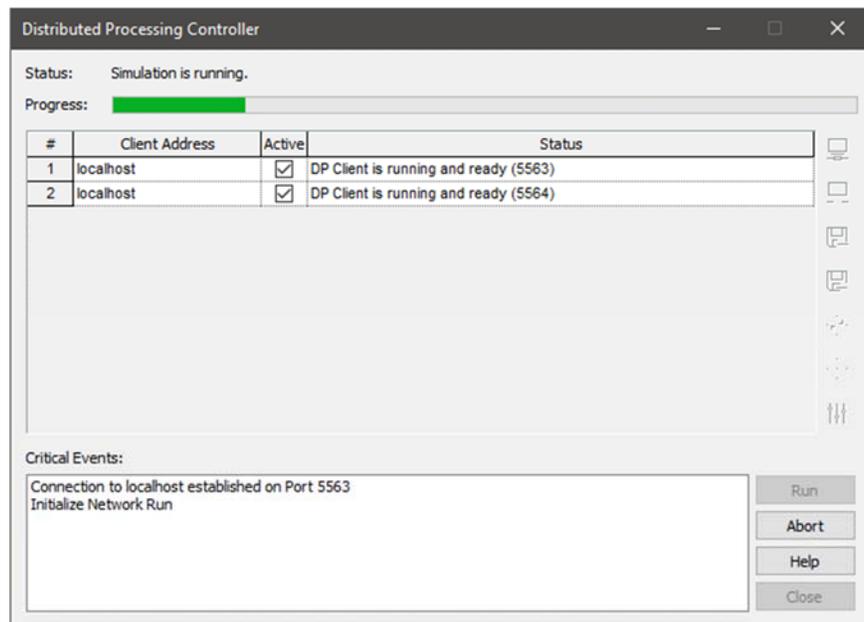


## Running a Distributed Processing Simulation

Pressing the **Update clients** button updates the status of all Clients.

Once you have connected to at least one Client process, you can run a simulation. To do so, press the **Run** button in the Controller dialog.

During the simulation, the Controller dialog will display the status of each Client process:



*The overall status of the simulation (based on the fraction of realizations that have been completed) is displayed at the top of the dialog. The status of each Client process (i.e., the realization currently being processed) is displayed under Status.*



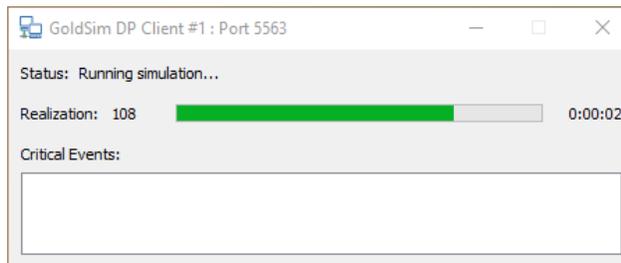
**Note:** When running a distributed processing simulation, the GoldSim Run Control Toolbar and Status bar are not used. The progress of the simulation is monitored from the Controller dialog rather than the main application status bar.

When a distributed processing simulation is started, the Controller process first transfers the model file (and any other required files) to each of the Client processes.

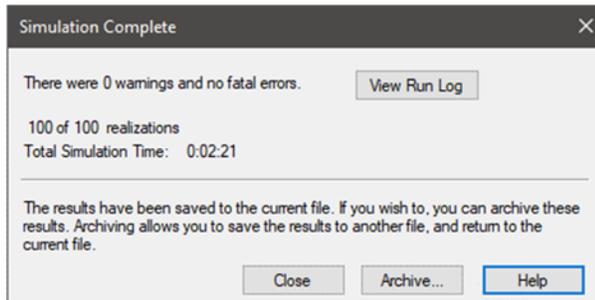
**Read more:** [Distributed Processing Auxiliary Files and Working Folders](#) (page 23).

Once the Client processes have received the required files, they can begin to receive and process realizations. The Controller assigns and sends individual realizations to each Client process, which then carries out its assigned realization, and then sends the results to the Controller process. The Client process is then assigned a new realization by the Controller. This continues until the specified number of realizations have been run.

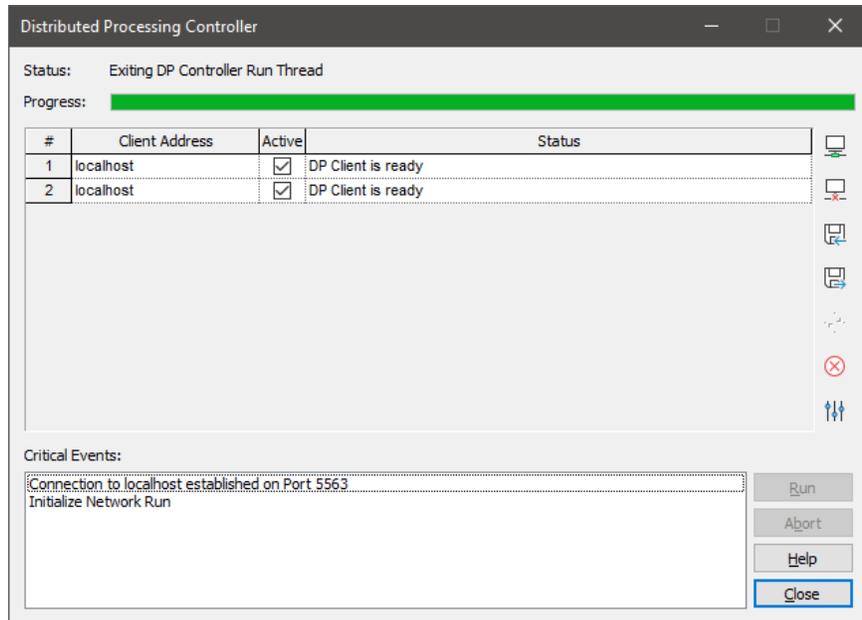
On machines with active Client processes, the Client dialog indicates the current status of the Client:



Once all of the realizations have been completed, the Controller process combines all of the results into a single GoldSim file (and displays a “Simulation Complete” dialog) just as if the entire simulation had been carried out on the Controller’s machine:



When you close the Simulation Complete dialog, the Controller dialog is still open, and looks like this:



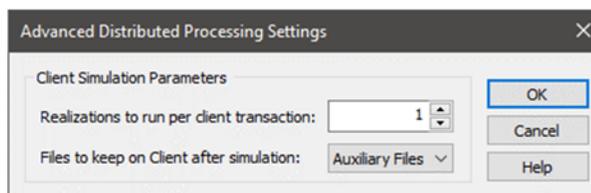
To browse the GoldSim model and view the results, press the **Close** button.

### **Assigning Multiple Realizations to Each Client Transaction**

When modeling systems with low-probability, high-consequence outcomes it is sometimes necessary to run a large number of realizations. When a realization can be run quickly the amount of time spent transferring information to Client processes for each realization individually can negate the performance benefits of distributed processing. As a result, GoldSim has an option to assign a block of realizations to a Client process. This option is accessed by pressing the **Additional Client Settings** button:



When you do so, the following dialog is displayed:



**Aborting a Distributed Processing Simulation**

A block of realizations can be assigned to a Client process by using the **Realizations to run per client transaction**.

By default the value is 1 (indicating a single realization will be assigned each time). The value can be increased to any number that is desired (e.g. specifying 5 means that the first Client transaction will assign realizations 1-5, the second 6-10 and so on).

While a distributed processing simulation is running, you may choose to abort the simulation before it is complete (i.e., in order to select a new group of computers to host Client processes). You can do this by pressing the **Abort** button in the Controller dialog.

When the simulation is aborted, The Simulation Complete dialog is displayed. Completed realizations are retained. All Client processes will be deactivated (but left running). When you close the Simulation Complete dialog, the controller dialog is displayed and all options are fully editable. When you close this dialog, if you subsequently press the **Update clients** button (to reactivate the Clients), the **Run** button is activated and is renamed **Run/Resume**.

If the **Run/Resume** button is pressed, GoldSim displays this dialog:



There are three buttons (in addition to Help):

**Resume Run.** This option makes use of realizations completed prior to the simulation being aborted, and runs the remaining realizations using the current list of Client processes.

**Start New Run.** This option discards all completed realizations and re-runs the entire simulation.

**Cancel.** Returns to the Controller dialog.

If the **Close** button is pressed in the Controller dialog, GoldSim assembles completed realizations up to the first gap in the realization sequence, and places the model into Result Mode. For example, if realizations 1,2,3,4, 6 and 7 were completed, only realizations 1,2,3, and 4 would be assembled and available in Result Mode.

**Activating, Deactivating and Terminating Client Processes During a Simulation**

Once you have started a distributed processing simulation, you can choose to activate an inactive Client process (e.g., if a machine becomes available) or deactivate a Client process that is currently active (e.g., if a user requires the machine for other work, or the machine’s performance is not sufficient).

You activate an inactive Client process by simply clicking on the **Active** checkbox. The Controller will automatically try to connect to the Client, and if successful, will begin to send it realizations.

**Read more:** [Connecting the Controller to the Clients](#) (page 12).

You can deactivate or terminate an active Client process by clearing the **Active** checkbox. The Controller will redirect the realization that the Client process was working on to another Client process, and will send no additional realizations to the Client process (unless it is subsequently reactivated).

Similarly, if a Client process is terminated during a simulation (i.e., if the Client process is closed, or if the machine hosting the process is turned off or rebooted), the Controller will redirect the realization that that Client process was working on to another Client process, and will send no additional realizations to the terminated process.



**Note:** If an active Client process is terminated (by first deactivating the Client in the Controller and then closing the Client dialog or turning off the Client machine) and then restarted during a simulation, the Controller will not automatically reconnect to the Client process and start to use it again. To reestablish the connection, you must check the **Active** checkbox for that Client process in the Controller dialog.

### **Distributed Processing Auxiliary Files and Working Folders**

When a distributed processing simulation is started, the first thing that the Controller does is to transfer the model file (and any other required files, such as spreadsheets referenced by Spreadsheet elements, DLLs referenced by External elements, and all File elements) to each of the Client processes.

The files are copied to a working folder on the machine hosting the Client process. The location of the working folder varies depending on the operating system. The best way to access the folder is to use variables provided by the operating system. Enter the following directly into the address bar for Windows Explorer:

`%ALLUSERSPROFILE%\GTG`

Within this folder will be a working folder called "DPClient $iiii$ ", where  $iiii$  is the port number. (If you run multiple Client processes on a single system, each port will have its own working folder). An index file (index.gst) is generated by GoldSim inside each working folder. This file stores information about any auxiliary files that have been saved in the folder.



**Note:** This is the default location for the folder. You can manually change this by editing the registry and creating a key called GoldSimDataFolder with the folder location under  
`HKEY_CURRENT_USER\Software\GTG\GoldSim\DistributedProcessing.`

If the GoldSim model uses External elements or External Pathway elements, copies of their DLL files will be created in the Client's local working folder.

If the GoldSim model uses Spreadsheet elements, then:

- If the associated spreadsheet file has a full pathname (e.g., "C:\test\file.xlsx") then it will be copied to the same path on the Client system under the working folder created by GoldSim (e.g., DPClient $iiii$ \test\file.xlsx).
- If the associated spreadsheet file has a local pathname (e.g., "testfile.xlsx") then it will be copied into the Client's working folder.
- If the associated spreadsheet file has a relative pathname (e.g., ".\xyz\testfile.xlsx") then it will be copied to a path on the Client system under the working folder created by GoldSim (e.g., DPClient $iiii$ \xyz\test\file.xlsx).



**Warning:** When doing distributed processing simulations, any referenced files (e.g., spreadsheets and DLLs) must be stored in the same folder as the GoldSim model file (or subfolders within that same folder). Relative paths to folders outside of the model folder should not be used.

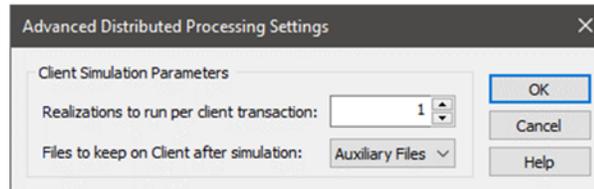
---

If you use File elements, the same rules discussed above for spreadsheet files apply to the File Element's local file name.

You can control whether or not the files in the working folders are saved at the end of the simulation. This option is accessed by pressing the **Additional Client Settings** button:



When you do so, the following dialog is displayed:



**Files to keep on Client after simulation** has three options:

**None:** All files generated by the Client process (as well as the working folder itself) are deleted.

**Auxiliary Files (default):** Any auxiliary files transferred to the Client process (e.g., DLLs, spreadsheets) are saved, but any intermediate files that might have been generated independently by the Client process (e.g., via DLLs) are not saved.

**All Files:** Any auxiliary files transferred to the Client process (e.g., DLLs, spreadsheets) are saved, *and* any intermediate files that might have been generated independently by the Client process (e.g., via DLLs) are also saved.



**Note:** The gsm file itself is never saved in the working folder.



**Note:** If a Client machine is turned off during a simulation, its working folders are not deleted.

While a distributed processing simulation is running, results sent back to the Controller are stored in temporary result files in the same directory as the model file. One file is created for each realization (or set of realizations), and named ModelFilename-RealX.gsr, where ModelFilename is the name of the model file, and X is the realization number (if a set of multiple realizations are saved, the filename becomes ModelFilename-RealX-Y.gsr, where X is the first realization in the set and Y is the last realization in the set). By default, these files are deleted when the Controller assembles all of the results into the model file.



**Note:** In some cases it is necessary to retain the temporary results files (e.g., to allow for recovery if the Controller process crashes). GoldSim provides a mechanism to do this by modifying a registry setting.

**Read more:** [Recovering a Simulation When the Controller Process Fails](#) (page 25).

### **Assigning Realizations to Multiple Clients at the End of the Simulation**

In some cases, when running a simulation with multiple realizations, it is possible that one or more of the realizations may “stall” or run very slowly for unanticipated reasons (e.g., due to hardware issues on the Client machine, or just because that particular Client is slow). In this situation, it is possible for all of the realizations except a small number to be complete. As a result, the simulation may take an extremely long time to finish (as it is waiting for the last few realizations to complete on the “slow” Clients).

To address this situation, toward the end of a simulation, if one or more Clients have no realizations assigned to them, you can instruct GoldSim to assign any incomplete realizations to the unused Clients. As a result, a realization can be assigned to multiple Clients. As soon as one of the Clients completes the realization, the other Clients working on the same realization are told to stop.

You can enable this behavior so that unfinished realizations are assigned to idle Clients by creating a registry DWORD called EnableReassignRealization in HKEY\_CURRENT\_USER\Software\GTG\GoldSim\Settings\ and setting it equal to a nonzero value. A value of 0 (or the absence of the the key) disables the behavior.



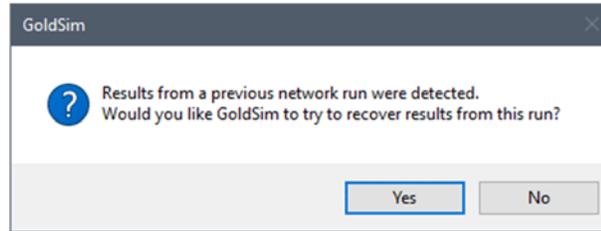
**Note:** This key is only meaningful if defined on the computer acting as DP Controller.

### **Recovering a Simulation When the Controller Process Fails**

Occasionally there will be situations where the computer hosting the Controller process crashes. If temporary result files are retained on the computer completed realizations can be recovered.

By default the temporary result files are processed (and deleted) as the simulation is running (to conserve disk space). To retain these files until the completion of the simulation, create a DWORD called DelayDeleteGSR in HKEY\_CURRENT\_USER\Software\GTG\GoldSim\Settings\ and set it equal to a value of 1.

When GoldSim is restarted and the model file that was being worked on is reopened, GoldSim will search for these temporary files when the **Run|Run on Network** option is selected. The following dialog will be displayed:



Pressing **Yes** will allow use of the completed realizations from the run where the Controller process crashed. When you enter the Controller dialog the **Run** button will be labeled **Run/Resume**. Clicking the **Run/Resume** button and then the **Resume Previous Run** option will run the remaining realizations needed to complete the simulation.

Pressing **No** will erase the temporary files from the interrupted run.

### **Distributed Processing Log Files**

When carrying out a distributed processing simulation, GoldSim creates ASCII log files for the Client processes. Each Client process has a log file named "DPClientLog $iiii$ .log", where  $iiii$  is the port number used by the Client. This file is located one level up from the Client's working folder (typically, C:\ProgramData\GTG).

Log files document the communication that takes place between the Controller and Client processes, and can be useful for debugging purposes.

## Limitations of the Distributed Processing Module

For the most part, if you need to run Monte Carlo simulations for a complex model, the use of the Distributed Processing Module is advantageous and strongly encouraged. There are some situations, however, in which use of the Distributed Processing Module may not be feasible.

In particular, because a distributed processing simulation requires that potentially large quantities of data (i.e., the GoldSim model file and other associated files, the results of each realization) be passed between the Controller and Client processes over a network, effective use of the Distributed Processing Module requires fast network connections (and fast computers). If the connections between the processes are slow, a distributed processing simulation may be impractical.

In addition, there are three situations in which distributed processing is not supported at all:

- You cannot run multiple scenarios (using the Scenario Manager) and utilize distributed processing. That is, the Scenario Manager provides no access to distributing processing capabilities.
- Because distributed processing can not be used for a single realization, and high resolution results (in which all unscheduled timesteps are displayed) are only available for single realizations runs, high resolution results can never be viewed for distributed processing simulations.

- You cannot export results to a Spreadsheet element when doing a distributed processing run (as each realization would result in a different spreadsheet file).

Scenarios, high resolution results and Spreadsheet elements are discussed in detail in Chapters 7, 8 and 10, respectively of the **GoldSim User's Guide**.



---

# Glossary of Terms

## **Client**

Also referred to as a Client process. A GoldSim executable running on a client machine that can be used to carry out distributed processing of a Monte Carlo simulation using the Distributed Processing Module.

## **Controller**

Also referred to as a Controller process. A GoldSim executable that manages Client processes in order to carry out distributed processing of a Monte Carlo simulation using the Distributed Processing Module.

## **DP Plus**

A premium version of the Distributed Processing Module that allows an unlimited number of Client processes to be used.



---

# Index

## A

- Aborting
  - distributed processing simulation 21
- Activating
  - Clients 21
- Adding
  - Clients for distributed processing 11

## C

- Client
  - closing 7
  - configuring firewall 9
  - connecting to 11
  - defined 7
  - importing and exporting 16
  - running GoldSim as a 7
  - terminating 7
  - terminating remotely 16
  - updating status 17
- Closing
  - Clients 7
  - Clients remotely 16
- Connecting to GoldSim Clients 11
- Context-sensitive Help 4
- Controller
  - defined 7
- Conventions
  - to describe key combinations 3
  - to describe mouse actions 3

## D

- Deactivating
  - Clients 21
- Distributed processing
  - aborting a simulation 21
  - activating/deactivating Clients 21
  - assigning multiple realizations 20
  - auxiliary files 22
  - closing a Client 7
  - importing/exporting Clients 16
  - limitations 25
  - log files 25
  - overview 7
  - recovering simulation after failure 24

- running a simulation 18
- starting a Client 7
- terminating Clients remotely 16
- updating Client status 17
- Distributed processing
  - connecting to Clients 11
- Distributed Processing Module
  - description 2
  - overview 7
  - steps for using 7
  - using 6
  - versions 2
- DP Plus 2

## E

- Email
  - support 4
- Exporting
  - Client lists 16

## F

- Files
  - used by Clients 22

## G

- GoldSim
  - blog 5
  - context-sensitive Help 4
  - Maintenance program 4
  - model library 5
  - online Help 4
  - running in Client mode 7
  - training 5

## H

- Help 4

## I

- Importing
  - Client lists 16

## K

- Key combinations
  - conventions to describe 3

## M

- Maintenance program 4
- Manual
  - Notes in 3

---

Warnings in 3  
Model library 5  
Mouse actions  
conventions to describe 3

## **R**

Realizations per Client transaction  
20  
Running  
distributed processing simulation  
18

## **S**

Support  
email 4  
GoldSim user forum 4  
model library 5

## **T**

Technical support 4  
Terminating  
Clients 7  
Clients remotely 16  
Training course 5

## **U**

Updating  
Client status 17