



GO-GLOBAL

for UNIX

Administrator Guide

For use with version 2.3.1

COPYRIGHT AND TRADEMARK NOTICE

Copyright © 1997-2016 GraphOn Corporation. All Rights Reserved.

This document and the software described in it are proprietary products of GraphOn, protected by the copyright laws of the United States and international copyright treaties. Any reproduction in whole or in part is strictly prohibited without the written consent of GraphOn. Except as otherwise expressly provided, GraphOn grants no express or implied right under any GraphOn patents, copyrights, trademarks or other intellectual property rights. Use of GO-Global for UNIX is subject to the terms detailed in the license agreement accompanying it. GraphOn, GO-Global, and the GraphOn logo are registered trademarks or trademarks of GraphOn Corporation in the United States and other countries. All other trademarks belong to their respective owners.

An XPM library is used in an unmodified form in accordance with the license granted from Bull Research France. Portions copyright © 1998-2000 The OpenSSL Project. All rights reserved. This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (www.openssl.org).

This software is based in part on the work of the Independent JPEG Group.

Open-source software: The GraphOn window manager (**gowm**) is based on software licensed under the GNU Library General Public License, version 2. In accordance with that license, the source code to the GraphOn Window Manager is available. Contact GraphOn if you would like to obtain machine-readable source.

Portions copyright © 1995-1998 Eric Young (eay@cryptsoft.com). All rights reserved. This product includes software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com). Portions of this software are copyright © 1996-2002 The FreeType Project (www.freetype.org). All rights reserved.

Information in this document is subject to change without notice. For the latest product documentation, including release notes and technical notes, visit our web site at <http://www.graphon.com>.

Printed in the United States of America.

CONTACT INFORMATION

GraphOn Corporation
51 East Campbell Avenue
Suite 128
Campbell, CA 95008
Toll Free: 1.800.GRAPHON
Phone: 408.688.2674
Fax: 408.626.9722

GraphOn
Venture House
Arlington Square
Downshire Way
Bracknell
RG12 1WA
United Kingdom
Phone: +44 1344.206549
Fax: +44 1344.206855

CHAPTER I - Introduction	4
<i>Introducing GO-Global for UNIX</i>	4
<i>Supported Platforms</i>	4
Server Platforms	4
Client Platforms	4
<i>System Requirements</i>	5
Server System Requirements	5
Client System Requirements	5
CHAPTER II - Server Installation	6
<i>Installing the GO-Global Server</i>	6
<i>Web-Enabling GO-Global</i>	7
Default Configurations and Directory Structure	8
<i>Configuring GO-Global</i>	8
Distribution	9
<i>Registration and Licensing</i>	9
<i>Distributing Client User Materials</i>	9
<i>Starting and Stopping the GO-Global Server</i>	9
<i>Uninstalling the Server Software</i>	10
<i>Reinstalling GO-Global for UNIX</i>	10
CHAPTER III - Server Configuration	11
<i>Server Configuration Options</i>	11
Non-Default Ports	11
Authentication Methods	12
goglobalux Script	12
<i>Using GO-Global in an NFS Environment</i>	14
Using NFS-Mounted Home Directories	14
Installing GO-Global into a Read-Only NFS Share	14
<i>Application Publishing</i>	14
GO-Configure	15
Publishing a New Application	16
Disabling Application Publishing by Client Users	18
Tuning Applications	18
Description	18
Paths	19
Optimizations	19
Display Options	21
Permissions	23
<i>License Management</i>	24
Establishing a Central License Server	24
Possible FlexNet Publisher Conflict	25
Stopping and Restarting GO-Global	25

CHAPTER IV - Client Installation	26
<i>Client Installation</i>	26
<i>Windows Clients</i>	26
Installing the Windows Client Software	26
Browser Client	27
Uninstalling the Windows Client Software	27
CHAPTER V - Using GO-Global Clients	28
<i>Overview</i>	28
<i>Connecting</i>	28
Authenticating	29
Starting a Session	29
Stopping or Suspending a Session	31
Resuming a Suspended Session	32
Multiple Connections	32
Closing a Connection	32
Running Browser Clients	33
Setting Proxy Values	33
<i>Copy and Paste</i>	34
<i>Printing (goprint)</i>	35
<i>File Transfer (goget/goput)</i>	35
<i>Status, Statistics, and Properties</i>	37
Connection Status	37
Connection Statistics	37
Application Properties	37
<i>Launching Applications from a Shortcut</i>	38
<i>Configuring the ActiveX Control</i>	40
CHAPTER VI - Session Configuration	41
<i>Session Configuration</i>	41
<i>Preferences</i>	42
<i>Command-Line Options</i>	43
<i>Working in the XDM Environment</i>	45
<i>Enhancing Session Performance</i>	46
<i>Enhancing Client Security</i>	46
CHAPTER VII – Troubleshooting	47
<i>Log Files</i>	47
Installation Logs	47
Master Log	47
Leader Log	48
Session Log	48
syslog	48
Xerrors	48

<i>Client Problem Situations</i>	48
<i>License Management Logging</i>	49
<i>FlexNet Publisher License Error Codes</i>	50
<i>Getting Help (Reporting a Problem)</i>	51
<i>Getting Help (Technical Support)</i>	56
CHAPTER VIII - Advanced Topics	57
<i>Certificates</i>	57
Certificate Authority	57
Server Key Certificate	58
<i>Obtaining a Trusted Server Certificate</i>	58
<i>Creating a Site Certificate Authority</i>	59
<i>Preliminary Considerations</i>	59
<i>Establishing the Certificate Authority</i>	59
<i>Creating a CA Key and Certificate</i>	60
<i>Creating and Signing Server Keys</i>	61
<i>Redundant License Servers</i>	63
Two-Server Redundancy	63
Three-Server Redundancy	63
License File Method	65
<i>Mapping a Locale</i>	65
<i>Mapping a Time Zone</i>	66
<i>Support for Modular Authentication (PAM)</i>	66
Disabling PAM	68
Index	69

Introducing GO-Global for UNIX

GraphOn's GO-Global for UNIX software is the thin-client, server-based solution that extends the reach of UNIX applications without requiring a client-side X server. GO-Global for UNIX enables enterprises, application service providers (ASPs), and independent software vendors (ISVs) to web-enable complex, feature-rich applications easily and cost-effectively for access over LANs, WANs, dial-up connections, or the Internet. GO-Global for UNIX delivers server-based UNIX applications to any display device with virtually no modifications and without requiring an X server installed on the display device. This guide will help you get your GO-Global software up and running quickly. It provides instructions for installing, configuring, customizing, and using the GO-Global Server and client. For the latest GO-Global documentation, including technical notes and version-specific release notes, please visit our web site at www.graphon.com/download/documentation.shtml.

Supported Platforms

Server Platforms

- Solaris 10 and 11 for SPARC and x86_64
- HP-UX 11.x for PA-RISC
- HP-UX 11.23 and 11.31 for ia64
- Red Hat Enterprise Linux v6.x and v7.x for x86_64
- SUSE Linux Enterprise Server v11 SP4 for x86_64 and ia64
- SUSE Linux Enterprise Server v12 for x86_64

Client Platforms

- Native Windows client for Windows XP, Windows 7, Windows 8, Windows 8.1, and Windows 10
- Firefox 31 ESR or later plug-in for Windows XP, Windows 7, Windows 8, Windows 8.1, and Windows 10
- ActiveX Control for Internet Explorer 11

Note: The web browser clients require that JavaScript be enabled.

System Requirements

Server System Requirements

To install and run the GO-Global Server software, your UNIX host must have at least the following resources available:

- Up to 180 MB disk space on the partition where the product will be installed (the installation program will tell you how much)
- A console X11R6 installation

Client System Requirements

Minimum	Recommended
200 MHz Pentium (or equivalent) processor	300 MHz Pentium II (or equivalent) processor
128 MB RAM	256 MB RAM
640 x 480 x 8-bit (256-color) display	1024 x 768 x 16-bit (65,536-color) display
4 MB free disk space	4 MB free disk space

Table 1.1

These memory and processor requirements correspond to the resources needed to run the initial X session. Required resources and installation requirements may vary among platforms. Moreover, the number of client users and the types of applications they run will ultimately determine the system load and resource demands of GO-Global.

The figures above are meant to provide safe guidelines to ensure that GO-Global does not exceed your system's resources. For more information and system-specific statistics, e-mail sales@graphon.com.

Installing the GO-Global Server

For simplicity, installation and configuration instructions refer to an environment variable called `$GOGLOBAL_ROOT`. This variable represents the directory in which you are installing GO-Global for UNIX. The default installation directory is `/usr/local/graphon`, but you may specify a different value during installation. When `$GOGLOBAL_ROOT` is used in examples elsewhere in this guide, substitute it with your actual installation directory.

The following steps outline the installation of the GO-Global software on your UNIX server. These instructions assume familiarity with the UNIX operating system. If you are uncomfortable using UNIX shell commands, consult your UNIX operating system documentation or system administrator.

1. Download the appropriate binary file from GraphOn's web site. For example: `GO-Global_for_UNIX-RedHat_Enterprise_Linux-7.0-x86_64.bin`
2. If you have copied the file from another server, ensure it is executable. For example:

```
# chmod u+x GO-Global_for_UNIX-RedHat_Enterprise_Linux-7.0-x86_64.bin
```

Note: When installing on HP11 in trusted mode, you may also need to specify a umask (e.g., "umask 000") before installing.

3. Login as root on the UNIX host.

```
# su
```
4. Execute the binary file. For example:

```
# GO-Global_for_UNIX-RedHat_Enterprise_Linux-7.0-x86_64.bin
```


5. On the **Enter Data** screen, specify values for each of the fields:
 - **GO-Global Directory:** The root directory for the GO-Global installation, referred to as `$GOGLOBAL_ROOT`.
 - **Non-Privileged FlexNet Publisher user:** This user is used to run the FlexNet Publisher license management system. This should be a non-superuser account. By default, this is the **flexlm** user.
6. You will be given the opportunity to restore a backup of a previous installation. If no previous installation exists, select **Don't Restore**.
7. The installer will also try to automatically obtain your license code. Use the Product Code that shipped with your GO-Global product. If your machine is unable to contact the GraphOn license server, follow the steps outlined in **Registration and Licensing** below.
8. GO-Global starts when installation is completed. To verify the software is running, examine the list of processes for the GO-Global Login Daemon, **gold**:

```
# ps -aef | grep gold
```

If you did not obtain a license, the software will not automatically start and will need to be manually started after you have obtained the license. To manually start the software execute:

```
# $GOGLOBAL_ROOT/etc/goglobalux start
```

Notes: If you are installing GO-Global on a system that previously did not support X display users, your system may need to be configured to allow more active processes and open files per user. This may require increasing the size of the process table and the file table for your UNIX system. You should also ensure that virtual memory has been specified in your system configurations and ensure that enough resources have been allocated for the number of supported clients.

This completes the minimum server installation.

If a license was not retrieved during installation, you can manually obtain your license using the **getlicense** utility. To start the utility execute: `# ${GOGLOBAL_ROOT}/etc/getlicense`. You will be prompted for contact information, presented with the GO-Global License Agreement, and prompted for your server's identification number. Once all information is entered, **getlicense** obtains the license and saves it to your machine. (To obtain a license, an Internet connection is required.)

Web-Enabling GO-Global

Web-enabling the GO-Global software allows end-users to use a web browser to initiate a GO-Global session. When configuring GO-Global for the web, the GO-Global host must have access to a machine with an HTML server. Because of the variety of web servers and the diversity of possible configurations, there is no *one* right way to web-enable GO-Global. Copy or symbolically link the contents of `$GOGLOBAL_ROOT/htdocs` to your web server's document root directory, or configure your web server to include these files for distribution.

When a user browses to *index.html*, GO-Global determines which web browser has requested data and returns the appropriate GO-Global software to the client machine.

Default Configurations and Directory Structure

The following instructions assume you are using the Apache web server and are familiar with how it is configured. Other web servers provide comparable structures. If you are not familiar with your web server's configurations, contact your system administrator or vendor for additional information.

Software	Default Directory
GO-Global	<code>\$(GOGLOBAL_ROOT)/htdocs</code>
Apache	<code>/usr/local/apache/htdocs</code>

If you have used directories other than the defaults, substitute these in the instructions below. GO-Global provides a client plugin for Mozilla Firefox and an ActiveX control for Internet Explorer. When these browsers open *default.html*, the appropriate browser-specific GO-Global software is automatically downloaded.

Configuring GO-Global

In order to web-enable the GO-Global host, the web pages created when you installed GO-Global need to be copied or linked from your GO-Global directory to your web server directory. There are two methods for achieving this. The first method requires creating a symbolic link. If your HTML server is not configured to enable symbolic links, you must use the second method.

Method 1: Creating a symbolic link

Assuming default installations of GO-Global and Apache, execute the following as root to symbolically link your GO-Global software to your web server:

```
ln -s /usr/local/graphon/htdocs
    /usr/local/apache/htdocs/goglobal
```

Method 2: Copying the contents of /usr/local/graphon/htdocs

Assuming default installations of GO-Global and Apache, copy the contents of `$(GOGLOBAL_ROOT)/htdocs` into `/usr/local/apache/htdocs/goglobal`.

To simplify administration, the creation of a "goglobal" subdirectory is recommended.

```
$ su
# cd /usr/local/apache/htdocs
# mkdir goglobal
# cd goglobal
# cp $(GOGLOBAL_ROOT)/htdocs/* .
```

The following is a complete list of files required for web-enabling GO-Global:

- IPviewer.html
- lplugin.cab
- NPinstall.html
- NPviewer.html
- Nplugin.jar
- default.html
- globals.js
- go_setup_2_2_x_xxx.exe
- goglobal_ux.cab
- index.html
- splash.gif
- docs (directory)

Distribution

To distribute GO-Global to end-users, provide one of the following:

- Instructions to browse to `http://[host_DNS]/goglobal/index.html`
- A web page with a link to `http://[host_DNS]/goglobal/index.html`

Registration and Licensing

The GO-Global Server will not run until you have registered your software and received a license. If the installer did not automatically retrieve your license during installation, follow these steps:

1. Obtain your Host ID:
`cat $GOGLOBAL_ROOT/etc/hostid.txt`
2. Obtain the Product Code that shipped with GO-Global.
3. Go to <http://license.graphon.com> to register your GO-Global software. Enter the **Product Code**, **Host Name**, and **Host ID**. If you are unable to register online, please e-mail licenses@graphon.com, or call 800-GRAPHON between 9 a.m. and 5 p.m. PST. Outside the U.S. and Canada, please call 408-688-2674.
4. Having completed registration, you will receive a data file called `license.dat`, which must be moved to your `$GOGLOBAL_ROOT/etc` directory.
`mv license.dat $GOGLOBAL_ROOT/etc`
5. After installing `license.dat`, GO-Global must be restarted. This is also true whenever `license.dat` is edited.
`$GOGLOBAL_ROOT/etc/goglobalux restart`

Distributing Client User Materials

The GO-Global Client suite is included in the `htdocs` directory, and can be used to distribute the appropriate clients directly to end-users. Native Windows and UNIX clients can be downloaded for native installations. Web-based installations only need to browse to the URL of the `htdocs` directory. The following table summarizes with examples:

Windows Client	<code>go_setup_version.exe</code>
UNIX Clients	<code>GO-Global_for_UNIX-platform.bin</code>
Browser-based clients	The URL of your <code>htdocs</code> directory, <code>http://host/goglobal/htdocs</code>

Table 2.1

When a client connects to the web server, the browser type is detected and the appropriate version of the client is used automatically.

Starting and Stopping the GO-Global Server

To start the GO-Global Server, issue the following command:

```
# $GOGLOBAL_ROOT/etc/goglobalux start
```

The start command executes the following processes:

- The GO-Global Login Daemon, `gold`

- The GO-Global License Manager Daemon, `lmgrd`

To stop the GO-Global Server, issue the following command:

```
# $GOGLOBAL_ROOT/etc/goglobalux stop
```

Uninstalling the Server Software

Uninstall the GO-Global for UNIX software by running `$GOGLOBAL_ROOT/uninstall`. Doing so erases all GO-Global for UNIX files. You will be given the opportunity to back up the following files:

Name	Function
<code>license.dat</code>	Enables connections and license management
<code>goglobalux</code>	The start/stop script with local host configuration data
<code>masterlog.conf</code> <code>leaderlog.conf</code> <code>sessionlog.conf</code>	The configuration files for the GO-Global Server's logging
<code>xmldb.conf</code>	The configuration for the GO-Global Server's database
<code>gold.conf</code>	Configuration options for the GO-Global Server

Table 2.2

You will be prompted for the directory where you would like to store these files. Provide the full path. The files will be saved in a subdirectory named `goglobal_save.date`.

Reinstalling GO-Global for UNIX

Before you reinstall or upgrade GO-Global for UNIX, uninstall your previous version to backup critical files and the GO-Global database. To do this, follow the uninstalling directions above.

During the reinstallation, you will be prompted to restore your saved files and database information. Restoring these files will enable you to continue to use the saved database and configuration files.

Some of the information contained in the database will contain paths to the installation directory. When restoring a saved database, you will need to perform the reinstallation into the same path as the uninstalled version was located.

Server Configuration Options

To perform authentication and manage sessions, GO-Global for UNIX uses the GraphOn Login Daemon (**gold**). On startup, gold reads up to four sources of configuration parameters. Options from each successive source override the corresponding values set in previous sources. Administrators are free to modify any configuration source, but it is recommended that changes be limited to the configuration file. The sources are read in the following order:

1. Environment Variables
2. `$GOGLOBAL_ROOT/etc/goglobalux startup script`
3. Command-line options
4. Options specified within `$GOGLOBAL_ROOT/etc/gold.conf`

Non-Default Ports

To set **gold**'s port to other than the standard of 491, edit the `tcpport` value in the `$GOGLOBAL_ROOT/etc/gold.conf` configuration file. Change the value to the new port number for GO-Global connections, then save the file. For example:

```
tcpport=492
```

Client users will have to specify the new port in the **Connection** dialog's **Server Address** field, or on the command-line when they log in, by following the server name with a colon and the new port number. For example:

myserver:492

Authentication Methods

GO-Global supports the following authentication methods;

- **pam** – supported on Solaris, Linux, and HP-UX 11. This method attempts to use the system's Pluggable Authentication Modules configuration to authenticate. This is usually the best option, and is the default.
- **pass** – supported on all platforms. This method attempts to authenticate directly against the system's `/etc/passwd` or `/etc/shadow` files.
- **aix** – supported on AIX only. This method uses the AIX LAM architecture to authenticate. This architecture is similar to PAM, but is an AIX-specific variant of it.

Authentication methods are specified via the `--auth` configuration option, documented below.

goglobalux Script

The **gold** daemon is run by the shell script `$GOGLOBAL_ROOT/etc/goglobalux`. This script is generated during installation and establishes the default operating parameters for the GO-Global Login Daemon, **gold** (`$GOGLOBAL_ROOT/bin/gold`). If you need to modify this script, please back up the original.

goglobalux declares the following shell variables:

Variable	Function
\$GOGLOBAL_ROOT	Full path to the GO-Global root directory. Default = <code>/usr/local/graphon</code>
\$FLEXNET_LOGFILE	Location of license manager log files. Default = <code>/dev/null</code>
\$GOLD_EXTRA_OPTIONS	Used to specify additional options to gold, if necessary.
\$FLEXNET_USER	Owner of the <code>lmgrd</code> and <code>graphon</code> licensing daemons. Default = <code>flexlm</code>
FLEXNET_SHELL	Specifies the shell used by <code>FLEXNET_USER</code> . This can be set here in the event that the script cannot automatically determine it, and is used to decide how to redirect the FlexNet Publisher output into the <code>FLEXNET_LOGFILE</code> .

Table 3.1

gold can take the following arguments:

Short Option	Long Option	Environment Variable	Function	Default
-h	--help		Prints usage and exits	
-v	--version		Prints version and exits	
-d	--daemon		Run gold as a daemon	false
-f	--forkmaintenance		Run gold maintenance in a child process	true
-p	--tcpport <i>port</i>	GOLD_TCP_PORT	gold TCP port number	491
-s	--sslport <i>port</i>	GOLD_SSL_PORT	gold SSL port number	791
-E	--no_explicit_suspend		Disables user-initiated suspending of sessions	Users may suspend sessions
-S	--suspend_age <i>minutes</i>	GOLD_SUSPEND_AGE	Maximum time sessions are allowed to be suspended before being terminated	-1 (infinite)
-M	--maintenance_interval <i>seconds</i>	GOLD_MAINTENANCE	How often maintenance should be performed	30
-T	--connectivity_timeout <seconds>	GOLD_CONNECTIVITY_TIMEOUT	Expire a connection if no communication within this interval (-1=never)	60
-A	--auth <auth_type>		Authentication method to use (pam, pass, etc)	pam
-I	--idle_timeout <minutes>	GOLD_IDLE_TIMEOUT	Expire a connection if no user mouse/keystrokes within this interval (0=never)	0
-P	--pid_path <path>	GOLD_PID_PATH	Use the specified path for recording process IDs	\$(GOGLOBAL_ROOT)/run/gold-\$(HOSTNAME).pid
-L	--lock_path <path>	GOLD_LOCK_PATH	Use the specified path for recording lock files	\$(GOGLOBAL_ROOT)/run

Table 3.2

Using GO-Global in an NFS Environment

GO-Global supports operating over NFS shares in two different ways: users' home directories can be located on NFS shares, and a single GO-Global installation can be shared across several hosts by installing it into an NFS share. In order to support these scenarios, changes will need to be made to the default GO-Global configuration.

Using NFS-Mounted Home Directories

The gold maintenance process runs periodically to perform routine cleanup tasks on users' personal GO-Global databases. This process runs as root, and must be able to write to the user database files. By default, these files are located in the user's home directory, as specified by the `userdb` parameter of `$(GOGLOBAL_ROOT)/etc/xmlldb.conf`. Please note that this configuration can cause significant performance problems on the server if the users' home directories are automounted. This path should be modified to point to a location that is writable by root as well as the user (owner) of the database. GraphOn recommends a directory with 1777 permissions. Note that if this location is shared among multiple GO-Global hosts, users will be prompted to resume sessions that were suspended on any host in the cluster regardless of which host they are actually connected to. (Verify that write access by root is allowed on the NFS server, as this is not the default NFS setting.) While this scenario is supported and will successfully resume the session, this behavior can be suppressed by including `$(HOSTNAME)` in the `userdb` specification.

For example, in `xmlldb.conf`, set `userdb = $(GOGLOBAL_ROOT)/db/$(USER).db`.
Next, create `$(GOGLOBAL_ROOT)/db/` and change its permissions to 1777.

Note:

Placing the master database onto an NFS share is not supported, even if the NFS share is writable. Simultaneous updates to the database can cause database to become corrupted.

Installing GO-Global into a Read-Only NFS Share

Once installed and configured, GO-Global can be run from a read-only NFS directory shared among multiple GO-Global hosts. To allow for this, changes to the default configuration are required. The master database will need to be moved to a location that will *not* be read-only, as specified by the `otherdb` parameter of `$(GOGLOBAL_ROOT)/etc/xmlldb.conf`.

There are also two path parameters in `$(GOGLOBAL_ROOT)/etc/gold.conf` that need to point to writable locations: `pid_path` (specifying the file into which the master gold's process ID is written); and `lock_path` (denoting a host-specific directory for lock files).

For example,

In `$(GOGLOBAL_ROOT)/etc/xmlldb.conf`, change `otherdb` to:
`/usr/local/etc/gold-$(HOSTNAME).db`

In `$(GOGLOBAL_ROOT)/etc/gold.conf`, change `pid_path` to:
`/usr/local/etc/gold.pid`

In `$(GOGLOBAL_ROOT)/etc/gold.conf`, change `lock_path` to:
`/var/tmp/goglobal/`

Note Placing the master database on an NFS share is not supported.

Application Publishing

The GO-Global installation process will scan the system for commonly used X11 applications, such as CDE, Gnome, KDE, xterm, etc. Any applications found on the system are automatically published and available to all client users.

Before client users can access a custom UNIX application directly, the GO-Global Server must be configured to publish the application using the GO-Global configuration utility, **GO-Configure**. Once published, the client will display the name of each application with an icon.

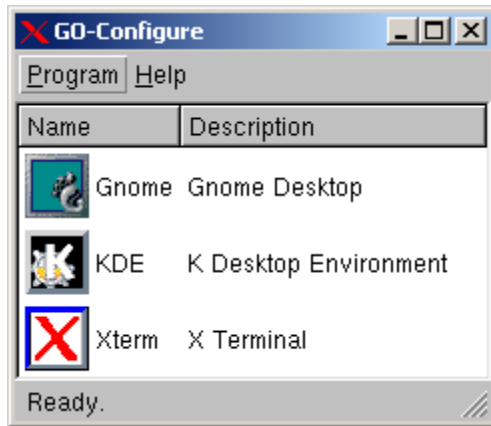
GO-Configure

The GO-Global application publishing database is stored in two parts. A system-wide database, consisting of all the applications that users are allowed to run, as determined by the system administrator, is published in `${GOGLOBAL_ROOT}/etc/${HOSTNAME}.db`. This database has UNIX file permissions that make it editable only by the root user.

Individual users each have their own personal database. This database maintains information about a user's running and suspended sessions, as well as any applications published by the user. Such published applications are only available to the specific user that published them. The user-specific databases are located in `$(HOME)/.graphon/$(USER).db`.

Practically speaking, this means that applications that are published in the system-wide database are only editable by the owner of that file, while individual users are free to publish their own applications in their personal databases. If a user tries to edit a published application in the system-wide database, they will be given the option to edit a "personal" copy of the item instead.

The database file locations can be modified by editing `$(GOGLOBAL_ROOT)/etc/xmldb.conf`.



GO-Configure can be run at the server's console, as follows:

```
# cd $GOGLOBAL_ROOT/bin
# ./go-configure
```

You can also launch GO-Configure from within a GO-Global client, as follows:

1. Connect to the server from the client.
2. Log in as **root** to publish to all users, or as a user to publish personal applications.
3. Double-click **GO-Configure**.

Publishing a New Application

The GO-Global Application Configuration program helps you publish a new application with its initial, essential configuration data.

To publish a new application

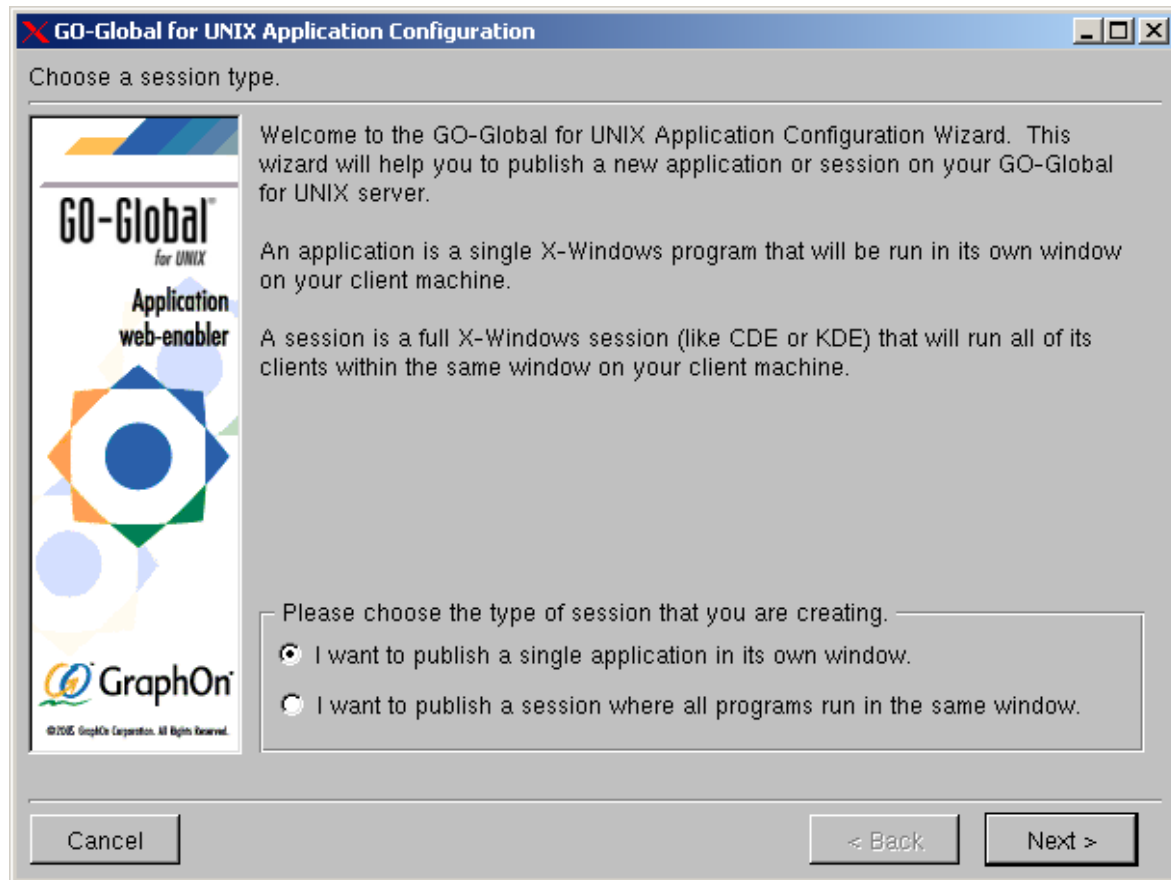
1. Launch **GO-Configure**, either from the UNIX command-line on the GO-Global Server or from within a GO-Global client.
2. Select Program, then New Item. The GO-Global Application Configuration program opens.
3. Choose the type of session you are creating:
 - A single application in its own window. (Also known as **multiple-window mode**.)
 - A session where all programs run in the same window. (Also known as **single-window mode**.)

Type of Session: Multiple-Window or Single-Window

In a **multiple-window** session, each application appears in an independent window. That is, the window is controlled by its own X window manager (an instance of GraphOn's GO Window Manager, **gowm**) and will not be dependent on any other application's session to stay open. This provides a more native feel for the applications, as they behave like applications run locally.

When running in multiple-window mode, it is possible for multiple applications to be running in the same session. While multiple-window mode is often desirable, it is important to remember which applications belong to a session, because closing the parent window will cause all the child windows to close, too. For example, if you were to launch a web browser from the command line in an **xterm** window, closing the **xterm** window would also close the web browser. All applications and windows belonging to a particular multiple-window session are listed with the session name `Multiple Window Session (gowm)`. To suspend one or more applications running in multi-window mode, look for an entry titled `Multiple Window Session (gowm)` rather than the name(s) of the application(s) in the list of **Running Sessions**.

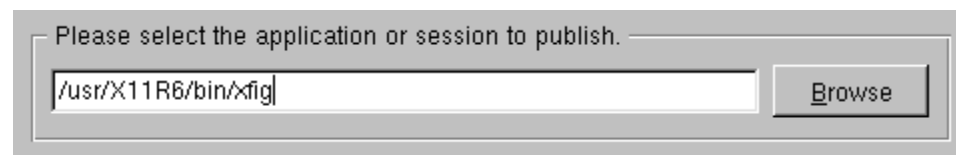
In a **single-window** session, all applications launched in the session appear in a single window, not in independent windows. This mode is ideal for remoting the entire UNIX desktop and simulating a console session. The CDE, Gnome, and KDE applications that come pre-published in GO-Global operate in single-window mode.



4. Click **Next**.
5. Specify the **Display Size**. (This screen applies to single-window sessions only. Otherwise, skip to step 7.) The size defaults to the client desktop, filling the available display. Otherwise, specify the height and width of the client application window.

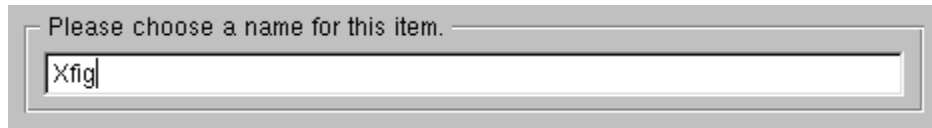


6. Click **Next**.
7. **Command Line**: Type the full path of the application to be published, or browse to it. Include any necessary command-line options.



8. Click **Next**.

9. Type a **Name** for your application. This will appear with the application icon in the user's list of published applications.



A screenshot of a graphical user interface dialog box. The dialog box has a title bar that says "Please choose a name for this item." Below the title bar is a single-line text input field. The text "Xfig" is entered into this field. The dialog box has a standard Mac OS-style border with a drop shadow.

10. Click **Finish** to complete the application-publishing procedure and close the GO-Global Application Configuration program.

Disabling Application Publishing by Client Users

If the administrator does not want client users to be able to publish applications for themselves, they must deny access to the GO-Configure application itself.

1. Login as root.
2. Launch GO-Configure.
3. Right-click the GO-Configure icon and select **Edit Item**.
4. On the **Permissions** tab, **Deny** permission to the users and/or groups that should not be able to publish applications.
5. Click **OK**.

Tuning Applications

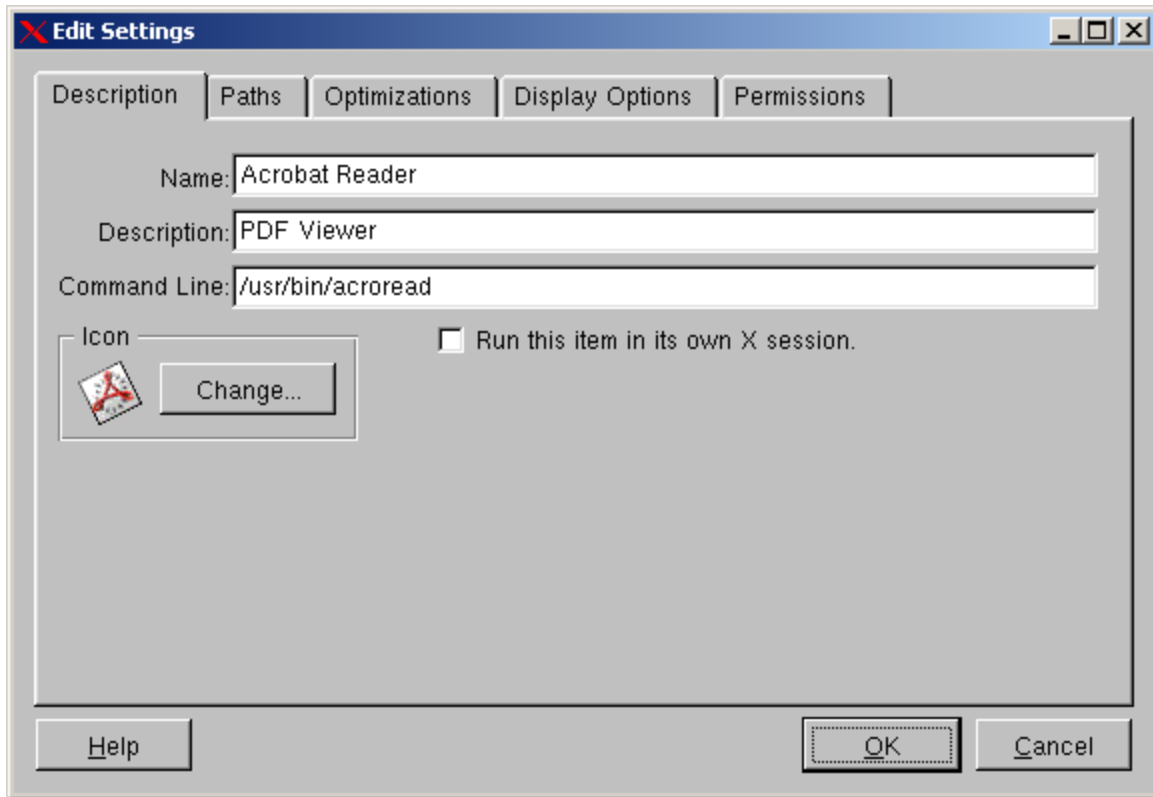
The preset configurations of an application published with the GO-Global Application Configuration program provide good, generic parameters. GO-Configure, the GO-Global configuration utility, can fine-tune published applications and sessions to enhance performance.

To edit an existing application

1. Connect to the GO-Global Server as the owner of the application to be edited.
2. Launch GO-Configure.
3. Right-click the icon of the application to be edited.
4. Select **Edit item**.
5. As necessary, edit the fields on the five tabs of the **Edit Settings** screen, as described below.

Description

- **Name:** The name of the published application, as displayed in the client's list.
- **Description:** A brief description of the application.
- **Command Line:** The UNIX command line that invokes the application. Include full path and any necessary command line options.
- **Icon:** The icon to be published with the application. It will appear with the Name in the client's list.
- **Run this item in its own X session:** See the section above titled, *Type of Session: Multiple-Window or Single-Window*.



Paths

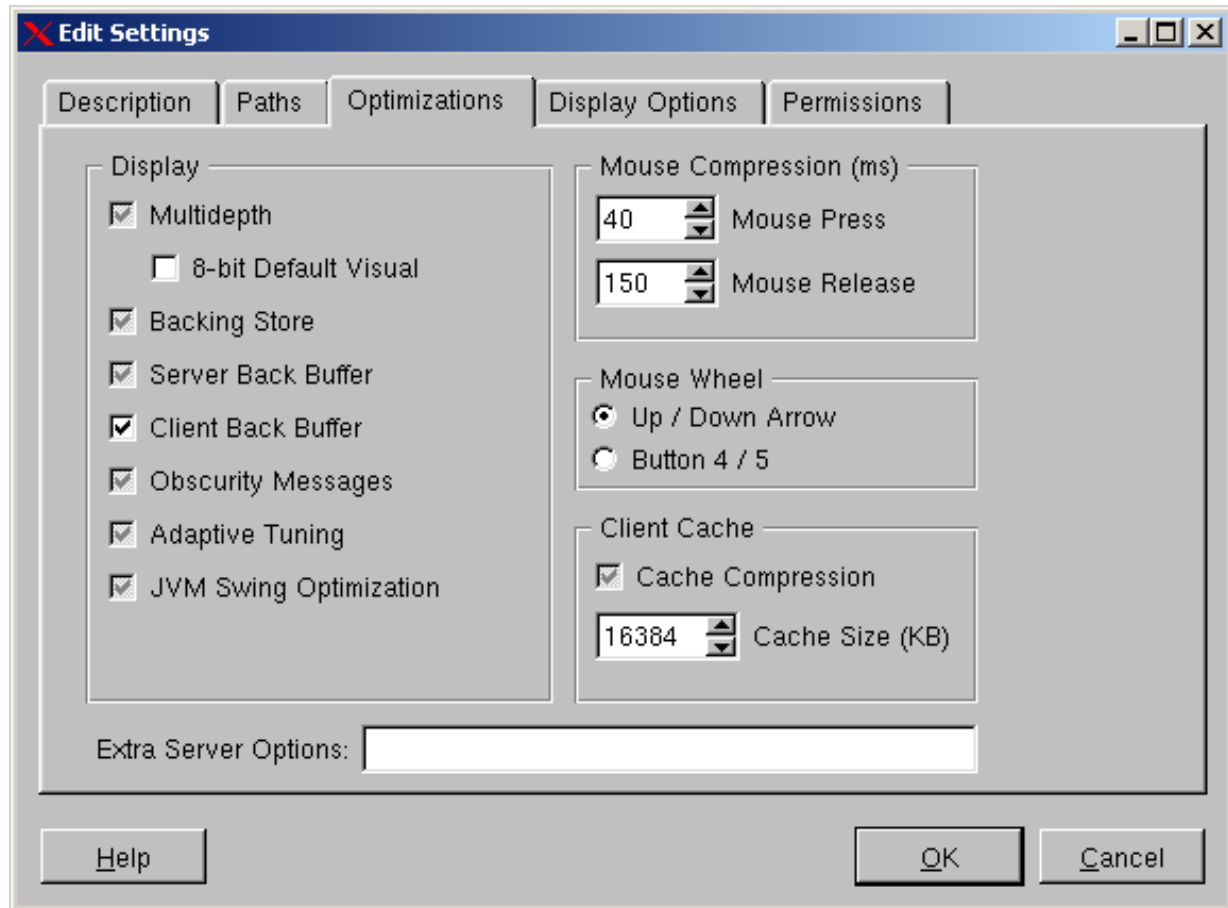
- **Prefixed/Appended Font Path:** Prefix or append X font directories to the font path. If a path contains a space, enclose it in double-quotes. Paths cannot contain commas.
- **Keytables Path:** The absolute path to the directory containing the region- and language-specific keyboard keytable files. Default = `$GOGLOBAL_ROOT/etc/keytables`.

Optimizations

The **Optimizations** tab has many checkboxes that have a “tri-state” functionality. When a checkbox shows an empty box, that feature is *disabled* and will never be turned on. When the checkbox shows a check on a white background, that feature is *enabled* and will always be turned on. When the checkbox shows a grayed-out checkmark, the feature is effectively *not specified* and may be turned on or off depending on the default settings. This state can be used to prevent the launch of an additional X server for each application when running in multiple window mode. For example, the application shown below requires that **Client Back Buffer** be enabled, but will run in any session, regardless of whether **Multidepth** (or **Backing Store**, etc.) are enabled.

- **Multidepth:** This option causes the server to provide both an 8-bit pseudocolor default visual and a 16-, 24-, or 32-bit true color visual simultaneously. The server can do this only for a native Windows client that is running a display of 16-bit color or higher. Off by default.
- **Backing Store:** Used to disable backing store and save-unders. Primarily for debugging. If a problem with the backing store support is suspected, it can be turned off. Leave On for general operation. On by default.

- **Server Back Buffer:** Maintains a server-side version of the client display. This option may slow the server but can increase the performance of certain applications reliant on this reverse data exchange. See the section titled, *Enhancing Session Performance*. Off by default.



- **Client Back Buffer:** Used for repainting and drawing correctly in 8-bit color. This option can slow drawing and is unnecessary over a LAN. On by default.
- **Obscurity Messages:** With this option on, when a session is completely covered or minimized, it tells the server to stop painting. This conserves bandwidth but also causes the client to pause when the session is unobscured, while the server repaints everything. On by default.
- **Adaptive Tuning:** Enables application configurations to be optimized for transmission based on the bandwidth and latency of the server-client connection. On by default.
- **Java Swing Optimization:** Optimizes the performance of Swing applications. Off by default.
- **Mouse Compression:** Determines how frequently in milliseconds the GO-Global client sends mouse data to the server when the mouse button is up or down. Lower numbers provide better responsiveness but consume more bandwidth. Defaults are 40 when the button is down and 150 when the button is up.
- **Mouse Wheel:** Configured differently depending on the application. **Up / Down Arrow:** Rolling the mouse wheel up acts as if the up arrow were pressed. Rolling the mouse wheel down acts as if the down arrow were pressed. **Button 4 / 5:**

Rolling the mouse wheel up sends a *Button4* message. Rolling the mouse wheel down sends a *Button5* message.

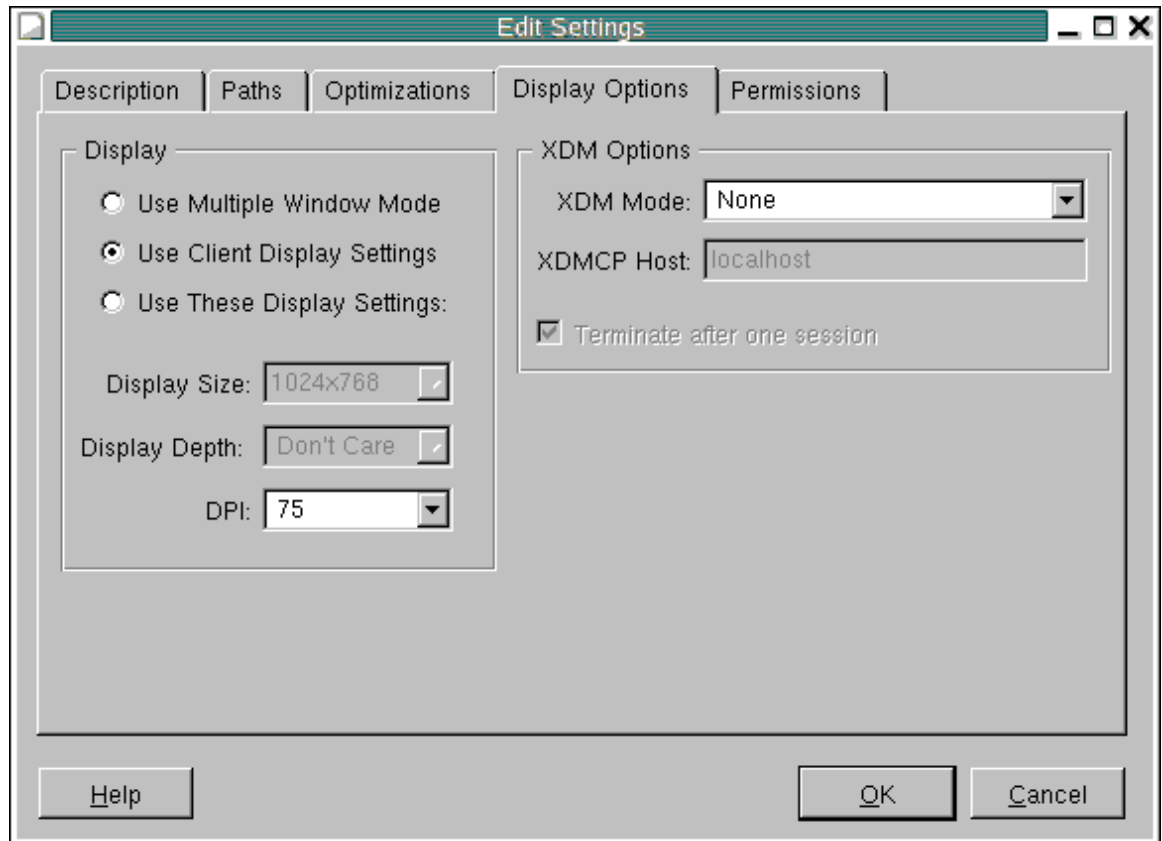
Tip:

Modern X applications treat *Button4* and *Button5* as scroll-wheel events, and when running such applications this is the preferred setting. Older applications that predate scroll wheels tend to work better when GO-Global maps the scroll wheel into Up/Down keypresses. Choose the setting that fits your application(s) best.

- **Cache Compression:** Determines whether data in the GO-Global client's cache is compressed. On by default.
- **Cache Size:** Specifies the maximum size of the client cache. 16 MB by default.
- **Extra Server Options:** Allows additional options to be called when the server publishes the application. None by default.

Display Options

- **Display:** Determines the mode in which the application is displayed on the client.
- **Use Multiple-Window Mode:** Application windows appear the size of the published application, with their own title bar and minimize, maximize, and close buttons.
- **Use Client Display Settings:** The settings of the client display device are used; particularly appropriate for remoting a desktop session.
- **Use These Display Settings:** The application will appear in a window with the dimensions defined by Display Size.
- **Display Size:** Maximum display dimensions of the application. Default depends on **Display** mode: **Multiple-window Mode** defaults to the size of the client's display, **Client Display Settings** defaults to full screen.



- **Display Depth:** Display depth, in bits, for the application. Defaults depend on the client's display depth setting:

Client Depth	Session Depth
8	8
16	24
24	24
32	24

Table 3.2

- **DPI:** Dots per inch of the display. Used in configuring the X server. Higher DPI provides higher resolution drawing but also consumes more bandwidth. Default is 75.
- **XDM Mode:** Selects mode of XDM login.

None: No XDM mode.

Direct: Directs an XDM request to the XDMCP host.

Indirect: Provides a list of all servers available to the XDMCP host, from which the client may choose a session.

Broadcast: Searches for all available XDMCP hosts, then initiates an X session and logs in to the first XDMCP host that responds.

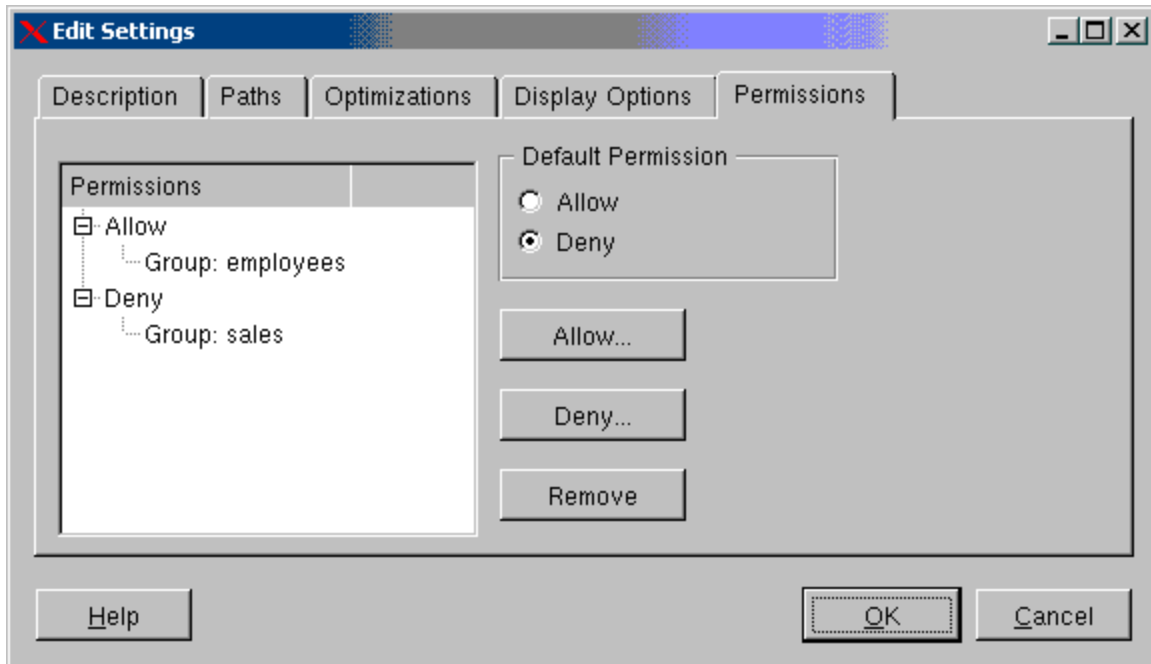
- **XDMCP Host:** The IP address or name of the XDM host to solicit for Direct mode.
- **Terminate after one session:** Terminates the connection after the first session exits.

Permissions

- **Default Permission:** Selects whether a user should be allowed or denied access to the item if there is no explicit permission listed for that user and if there are no explicit permissions listed for any of the groups to which the user belongs. The default permission is also used when a user belongs to multiple groups that have conflicting permissions.
- **Allow:** Add a user or group that has permission to use the application.
- **Deny:** Add a user or group that does not have permission to use the application.
- **Remove:** Deletes the selected user or group from the Add or Deny category.

To publish an item to one or more specific users or groups, denying all others, grant the **Allow** permission to the desired user(s) and/or group(s) and set the default permission to **Deny**.

The permissions model allows for some complex scenarios. In the following example, an organization's system includes six groups: consultants, employees, sales, marketing, engineering, and admin. Users belong to either the consultants group or the employees group. Users also belong to a department group (either sales, marketing, engineering, or admin). To publish an item to all employees (excluding consultants), but not allow members of the sales group to access it, set the default permission to **Deny**, grant employees the **Allow** permission, and grant sales the **Deny** permission. Even though some of the users in the sales group belong to the employees group, they will be assigned the default permission (**Deny**, in this case) because they belong to groups with conflicting permissions.



License Management

GO-Global uses Flexera's **FlexNet Publisher** to provide license management. For most configurations, licensing should be invisible after the license file is installed during registration. Certain sites or configurations may require implementation of specialized licensing schemes to meet their needs. The following guidelines are not exhaustive.

For more information on FlexNet Publisher, refer to the *License Administrator Guide*, available at http://www.globes.com/support/fnp_utilities_download.htm#downloads

Establishing a Central License Server

By establishing a central license server, you enable multiple servers to make use of a single license.

To establish a central license server

1. Choose a machine to act as the license server.
2. Install the FLEXnet Publisher license on the license server as
\$GOGLOBAL_ROOT/etc/license.dat.
3. Make a copy of license.dat. Name it client.license.dat.
cp license.dat client.license.dat
4. Edit client.license.dat, replacing the VENDOR, INCREMENT, and HOSTID lines with
USE_SERVER. The resulting file should read:

```
SERVER hostname.domain.com hostid
USE_SERVER
```

(The SERVER line must be exactly the same as in license.dat on the license server.)

5. Use **ftp** to copy `client.license.dat` to `$GOGLOBAL_ROOT/etc/license.dat` on each GO-Global Server that will need access to the license. For example:

```
# ftp -u goglobal otherserver1
ftp> (provide goglobal's password)
ftp> cd /usr/local/graphon/etc
ftp> put client.license.dat license.dat
ftp> close
ftp> open otherserver2
(etc.)
```

If a license was not retrieved during installation, you can manually obtain your license using the **getlicense** utility.

To start the utility execute: `# ${GOGLOBAL_ROOT}/etc/getlicense.`

You will be prompted for contact information, presented with the GO-Global License Agreement, and prompted for your server's identification number. Once all information is entered, **getlicense** obtains the license and saves it to your machine. (To obtain a license, an Internet connection is required.)

Possible FlexNet Publisher Conflict

If other server programs or applications use **FlexNet Publisher** licensing manager, you may not be able to activate your GO-Global license. To circumvent this problem, you may specify a unique port number for your GO-Global license server. At the end of the `SERVER` line in `license.dat`, add a space and the number of the license port to be used. For example:

```
SERVER hostname.domain.com HOSTID 27009
will establish a GO-Global license server using port 27009.
```

Stopping and Restarting GO-Global

Any time you modify `license.dat` while GO-Global is running, you will need to stop and restart GO-Global by issuing the following command:

```
# $GOGLOBAL_ROOT/etc/goglobalux stop
# $GOGLOBAL_ROOT/etc/goglobalux start
```

Client Installation

The GO-Global client software can be installed and run in two ways:

- Windows native client — download and install
- Windows and UNIX browser client (plug-in)

Windows Clients

As the administrator, you need to decide how you are going to distribute the client software, as well as how you will distribute future upgrades. The native clients (installed or plug-in) take advantage of the operating system of the client computer and should be used. The following questions need to be answered:

- Can your users install or use the client from the web?
- Can your users do their own installations and also reinstall upgrades?

	Requires a web server	Requires user to install client and upgrades
Standard Windows Client	No	Yes
Browser Client	Yes	No

Table 4.1

Installing the Windows Client Software

GO-Global client software is downloaded or copied to the client via **FTP** or some other means. An installation program provides step-by-step instructions for installation and setup. The default installation directory for the client software is `C:\Program Files\GO-Global for UNIX v2.3\`.

To install the Windows Client Software

1. Download the software.
2. Run `go_setup_version.exe`.
3. Select **Run** from its current location.
4. Follow the instructions of the installation program.
5. Click Start | Programs | GraphOn | GO-Global for UNIX 2.3 | GO-Global for UNIX.

Browser Client

The browser client opens a GO-Global connection without installing any permanent client software. When making a connection to a GO-Global Server via a browser, the particular client implementation depends on your specific browser. For example, if you connect using Internet Explorer, the ActiveX Control is loaded. Mozilla users will see the GO-Global Mozilla Plug-in.

Uninstalling the Windows Client Software

If you installed the GO-Global client with the web installer or downloaded it, an uninstall program is part of your installation.

To uninstall the Windows Client Software

1. On the Windows **Start** menu, select **Settings**, then **Control Panel**.
2. Double-click Add/Remove Programs.
3. From the list of installed programs, double-click **GO-Global for UNIX 2.3.0**.
4. Select **Remove**, then click **Next**.

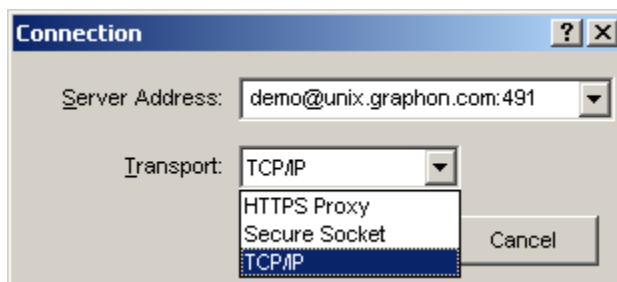
Overview

A GO-Global *connection* links the client to the server. Once *authenticated*, a user can start a *session* by launching applications or desktop environments. The following overview outlines the steps to connect and start a session.

1. **Launch the client.** Depending on how the client was installed and the type of client, you may need to double-click its icon on the **Desktop**, select it from the **Start** menu, or connect to a certain web page.
2. **Connect to the server.** Enter the **Server Address** (host name or IP address) of the server, and click **Connect**.
3. **Authenticate.** Type a **Username** and **Password** for the server, and click **OK**.
4. **Start a session.** Choosing from the displayed list of published applications, double-click the application's icon or right-click it and select **Launch**.

Connecting

GO-Global uses a network connection between the client and server for servicing protocol requests. To open a new connection, go the **Connection** menu and select **Open**. The **Connection** dialog appears. A list of recent connections is preserved on the client machine and available in the drop-down list. The same list is available by going to the **Connection** menu and selecting **Recent Connections**.



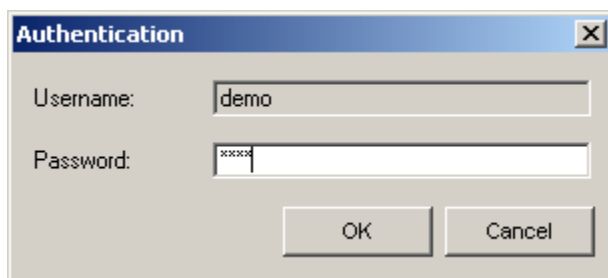
The **Server Address** is either the host name by which the machine is identified on the network or the numeric IP address of the host machine. To establish a GO-Global connection using the standard port (491) and standard transport mode (unencrypted TCP/IP), only the host name is required.

Over each connection, you can run one or more sessions and you can open additional connections, whether to the same or to different GO-Global Servers.

Authenticating

As part of establishing a connection, the server requests authentication credentials. Typically, these are in the form of a username and password. For specific information on what is required with your implementation, contact your system administrator.

1. Type your **Username** and **Password**.
2. Press **Enter** or click **OK** to proceed.



By default, username and password are *not* saved on the client machine, because this would allow bypassing of the **Authentication** screen.

Starting a Session

Upon establishing a connection to a GO-Global Server, a list of available applications will be displayed in the main window. These may include:

- X desktop sessions
- X applications
- The GO-Configure application publishing utility



Launching any of these applications starts a GO-Global *session*. Basically, a session is an instance of an application running over a connection. There are two types of session: **multiple-window** and **single-window**.

To start a session (launch an application), double-click the application's icon or right-click the icon and select **Launch**.

The **Connection** menu yields the following information on server connections and sessions:

- **Current Connections:** Connections that have been opened to servers and are currently running. Each connection is represented by a tab between the menu bar and the list of published applications.
- **Recent Connections:** Lists the last five server connections made from this client.
- **Running Sessions:** Lists all sessions that are currently active between the client and all connected servers.
- **Suspended Sessions:** Lists all sessions that have been opened and subsequently suspended. Though these are not active (running) sessions, neither are they stopped sessions. They can be resumed at any time without having to relaunch the associated application(s).

Note: Application icons can be dragged from the client window to the desktop, creating a shortcut with all the associated launch parameters.

Stopping or Suspending a Session

When you want to end a session, you will have the option of either *stopping* it (terminating both the display running on the client and the application running on the server) or just *suspending* it (terminating the display running on the client but leaving the application running in a “sleeping” state on the server).

To stop or suspend a session

1. On the **Connection** menu, click **Running Sessions**, and then click the identifier of the session you want to stop or suspend. The following dialog appears:



2. Make your selection:
 - **Suspend** disconnects the client session from the X server, but the associated processes continue to run on the GO-Global host. Suspended sessions can be resumed at any time. This is especially useful if you need to shut down the client but want to resume the session where you left off. All windows associated with a session will be suspended; none will be stopped. For example, if you are running a CDE session, when you resume it, all the applications within it that were running will reappear just as you left them when you suspended the session.

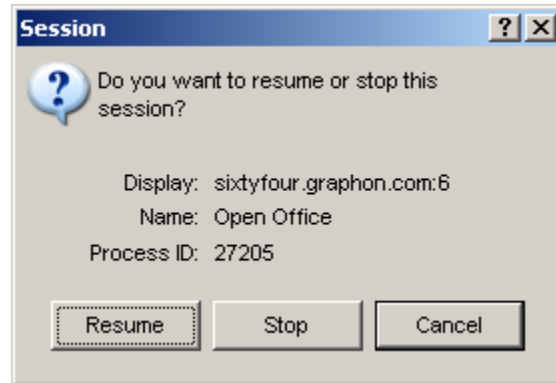
Note: The contents of the server clipboard are not preserved when suspending a session. The contents will be lost as if the session were stopped.

- **Stop** closes the client session and closes all X applications and associated processes on the GO-Global host. It may also close other processes that are running in conjunction with the closed client session. Note that all windows associated with a session are stopped.
- **Cancel** simply dismisses the dialog and returns to the main window with no affect.

Resuming a Suspended Session

To resume a suspended session

1. On the **Connection** menu, click **Suspended Sessions**, and then click the identifier of the session you want to resume. The following dialog appears:



2. Make your selection:
 - **Resume** reconnects the client session to the X server, and the session reappears just as it did when you suspended it.
 - **Stop** closes the client session and closes all X applications and associated processes on the GO-Global Server. It may also close other processes that are running in conjunction with the closed client session.
 - **Cancel** simply dismisses the dialog and continues to the next dialog or to the GO-Global client main window, without affecting any suspended sessions. If you select Cancel for any session, you can still resume that session later.

You will get this dialog for each suspended session. Make a selection for each session.

Note: You will also be prompted to resume sessions upon reconnecting to a server where you have suspended sessions.

Multiple Connections

Multiple connections may be created from a single client. These connections may be to the same or different GO-Global Servers, but if they are to the same server, different usernames must be used when making the additional connections.

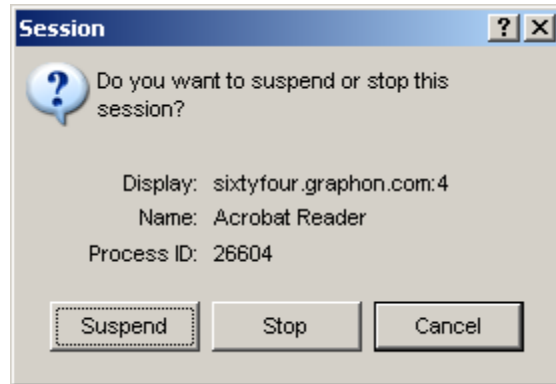
To move between servers, click the tab representing the server you want to access.

Closing a Connection

Closing a GO-Global client connection stops the transfer of all data between the client and server. It does not necessarily stop the application running on the server. Upon closing a connection, you are prompted to either stop or suspend all of the sessions that are running over that connection.

To close a connection

1. Click on the tab representing the connection you want to close.
2. On the **Connection** menu, select **Close**. The following dialog appears:



3. Make your selection:
 - **Suspend** disconnects the client session from the X server, but the associated processes continue to run on the GO-Global host. Suspended sessions can be resumed at any time. This is especially useful if you need to shut down the client but want to resume the session where you left off. All windows associated with a session will be suspended; none will be stopped. For example, if you are running a CDE session, when you resume it, all the applications within it that were running will reappear just as you left them when you suspended the session.
 - **Stop** closes the client session and closes all X applications and associated processes on the GO-Global host. It may also close other processes that are running in conjunction with the closed client session. Note that all windows associated with a session are stopped.
 - **Cancel** simply dismisses the dialog and returns to the main window with no affect.

Running Browser Clients

There is no installation necessary for the web browser client on Windows. This client opens a GO-Global connection without installing any permanent client software.

1. Open a web browser.
2. Enter the GO-Global Server's URL, provided by the system administrator, in the browser's **Location** or **Address** field, and press **Enter**.
3. When prompted to grant permission, do so, provided the request specifies GraphOn Corporation or another familiar source.

Setting Proxy Values

If your LAN uses a firewall to shield it from the Internet, your web browser may need to go through a proxy server before accessing the Internet. A proxy server is a computer on your LAN that provides additional security between your computer and the Internet. If your LAN uses a proxy server to connect to the Internet, you need to provide your web browser with the name of the proxy server and the number of the port at which your browser needs to connect to it.

Ask your system administrator whether you need to set proxy values for your browser. If you do, enter the information according to the following instructions.

In Mozilla Firefox

1. From the **Edit** menu, click **Preferences | Advanced | Proxies**.
2. Select one of the following options:
 - If you don't have to go through a proxy, select **Direct Connection to the Internet**.

- If you do have to go through a proxy, and your system administrator has set up a proxy configuration file, select **Automatic proxy configuration**, then enter the URL.
- If there is a proxy server, but there is no proxy configuration file, select **Manual proxy configuration**, then click **View**. There may be more than one proxy server. For each network service (HTTP, FTP, etc.), type the name or IP address, and port number, of the server running proxy software. Under **Exceptions**, type the names of any domains or servers that you can connect to directly, bypassing the proxy (for example, your company's internal domain or specific computers on your LAN). Use commas to separate names.

In Internet Explorer

1. From the **Tools** menu, click **Internet Options | Connections | LAN Settings**.
2. Select one of the following options:
 - If you don't have to go through a proxy, check only the **Automatically detect settings** box.
 - If do have to go through a proxy, and your system administrator has set up a proxy configuration file, check only the **Automatic configuration script** box, then enter the **Address** (URL).
 - If there is a proxy server, but there is no proxy configuration file, select **Use a proxy server**, then enter the **Address** (name or IP address) and **Port** number of the proxy server. If different proxy servers are used for different network service (HTTP, FTP, etc.), click **Advanced**, then enter the appropriate addresses and port numbers.
 - If there are any domains or servers that you can connect to directly, bypassing the proxy (for example, your company's internal domain or specific computers on your LAN), type their names under **Exceptions** (on the **Advanced** screen). Use commas to separate names.

Copy and Paste

GO-Global supports copying plain text between the server and the client.

To copy text between a GO-Global session and a local application

1. In your UNIX document, select the text you want to copy. (In some cases, this is enough to place the text in the copy buffer, and you may skip to step 3.)
2. Select **Edit | Copy**.
3. In the target document, click the cursor where you want the text placed.
4. Paste the text using the local application's paste function.

To copy text between a local application and a GO-Global session

1. In the document containing the text you want to copy, select the text you want to copy.
2. Copy the text using the local application's paste function.
3. In the target UNIX document, click the cursor where you want the text placed.
4. Select **Edit | Paste**.

Printing (goprint)

By default, UNIX applications accessed via a GO-Global session will print to a printer on the server, assuming the server has a printer configured. GO-Global supports client-side printing with the GraphOn printer utility **goprint** (`$GOGLOBAL_ROOT/bin/goprint`). If you want to print on the client's printer, specify **goprint** in your print command.

goprint can also be used from the UNIX command prompt, as follows:

Command Syntax

```
goprint {file}
```

Options

file...	Specifies the file(s) to be submitted to the goprint printer queue. Files to be printed must be in a format recognized by your printer.
	If no file is specified goprint will read from standard input.

Table 5.1

Examples:

A. Print an ASCII file `readme.txt` (with line ending translation)

1. Open an X terminal window.
2. **cd** to the directory containing the file to be printed.
3. Enter the command to print the file:
`cat readme.txt | unix2dos | goprint`
4. The **Print** dialog box appears on the client. Select the desired printer to send the document to.

B. Print a `.jpg` file from a Linux machine

1. Open an X terminal window.
2. **cd** to the directory containing the file to be printed.
3. Format the jpeg for a postscript printer by typing:
`a2ps -o- image.jpg | goprint`
4. The **Print** dialog box appears on the client. Select the desired printer to send the document to.

File Transfer (goget/goput)

Files can be transferred between the GO-Global Server and the client with the GraphOn file transfer utilities, **goget** and **goput** (`$GOGLOBAL_ROOT/bin/goget` and `$GOGLOBAL_ROOT/bin/goput`). These utilities are used at the UNIX command prompt to initiate the file transfer operation. The **goget** utility will “get” a file from the client and the **goput** utility will “put” a file from the host to the client.

Command Syntax

```
goget [-hvpl] [clientfile] [hostfile | -]
```

Options

<code>-h, --help</code>	Print the usage information.
<code>-v, --version</code>	Print the version of the utility.

<code>-p, --progress</code>	Print a “#” character for each block of data transferred. Default is false.
<code>-l, --loglevel <level (0-7)></code>	The level at which messages should be displayed. (1=fatal, 2=critical, 3=error, 4=warning, 5=note, 6=debug, 7=trace). Default is 4.
<code>clientfile</code>	The file to retrieve from the client. If none is specified, the user is prompted. Default is prompt user.
<code>Hostfile</code>	The filename to save the file to on the host. If none is specified the file is written to stdout. (Use "-" for stdout when specifying a clientfile.)

Table 5.2

```
goput [-hvxpl] [hostfile | -] [clientfile]
```

Options

<code>-h, --help</code>	Print the usage information.
<code>-v, --version</code>	Print the version of the utility.
<code>-x, --execute</code>	Executes the file after transfer is complete. This is dependent on an application being configured to execute the file in the client environment. For example, after transferring a .txt file to a Windows-based client and using the -x option, the file will be displayed automatically by Notepad. Default is false.
<code>-p, --progress</code>	Print a “#” character for each block of data transferred. Default is false.
<code>-l, --loglevel <level (0-7)></code>	The level at which messages should be displayed. (1=fatal, 2=critical, 3=error, 4=warning, 5=note, 6=debug, 7=trace). Default is 4.
<code>hostfile</code>	The file to retrieve from the host. If none is specified, stdin is used. (Use "-" for stdin when specifying a clientfile.)
<code>clientfile</code>	The filename to save the file to on the client. If none is specified the user is prompted. Default is prompt user.

Table 5.3

Note: `goput -x` works on the Windows Client only.

Examples:

A. Transfer the file `readme.txt` from the current directory on the GO-Global Server to the client.

1. Open an X terminal window.
2. `cd` to the directory containing the file to be transferred.
3. Enter the command to copy the file:
`goput readme.txt`
4. When the **Save As** dialog box appears, browse to the directory where you want to write the file. Provide a name for the file.
5. Click **Save** to transfer the file.

B. Transfer the file `readme.txt` from the directory `C:\Program Files\Star Office` to the GO-Global Server.

1. Open an X terminal window.
2. Enter the command to copy the file:
`goget c:\\program\\files\\star\\office\\readme.txt > readme.txt`
3. The file is transferred.

—or—

1. Open an X terminal window.
2. Enter the command to copy the file:
`goget > readme.txt`
3. The **Open** dialog box appears. Browse to and select the file to be copied. Click **Open**.
4. Click **Save** to transfer the file.

Status, Statistics, and Properties

The client application provides useful information that lets users view a session's status, a connection's statistics, and an application's properties.

Connection Status

The client status of a highlighted application can be viewed in the status bar of the client window by selecting **View** and clicking **Status** bar.

Connection Statistics

Connection statistics provide useful information about the connection between the server and client. The **Statistics** display provides the total bytes received from and sent to the server. It also provides the transfer rate in bytes per second. To open the **Statistics** bar, click **View**, then **Statistics**.

Application Properties

Each application has a list of properties that describes how the application is published to the client. This configuration data is stored in the GO-Global database on the GO-Global host. To view an application's properties, right-click the application's icon and select **Properties**. Application properties are described below. Defaults are given where appropriate.

- **Description:** A brief explanation of the application.
- **Path:** The full path of the application on the server.
- **Runs in its own session:** The application runs in a window independent of all other X sessions.
- **XDM:** Which mode of XDM the application is configured for, if any. Default = None.
- **Icon path:** The full path of the icon published with the application.
- **Multiple-window mode:** If Yes, the application and any associated applications appear in their own windows.
- **Display size:** The width and height of the session window on the display device. Default = the size of the application in multiple-window mode.
- **Display depth:** The color depth of the session. Client-dependent.
- **Appended font path:** Additional font paths which follow those of the host.

- **Prepended font path:** Additional font paths which precede those of the host.
- **Adaptive tuning:** Enables application configurations to be optimized for transmissions, based on the quality of the server-client connection. Default = On.
- **Swing Optimization:** Whether optimizations for Java Swing applications are used. Default = Off.
- **Multidepth:** Whether multiple color depths are enabled on the server. When enabled, windows with depths other than the depth of the root window (always 8-bit) can be created. For example, the root window would be 8-bit but a secondary window could be 24-bit. Default = Off.
- **Server back buffer:** Whether the server back buffer is enabled, allowing client painting data to be retained. Default = Off.
- **Client back buffer:** Whether the client back buffer is enabled, allowing improved 8-bit display and redrawing of applications. Default = On.
- **Backing Store:** Disables backing store and save-unders. Generally used only for debugging. Default = On.
- **Obscurity messages:** Whether the client informs the server when an application is completely covered. Conserves bandwidth. Default = On.
- **Mouse compression:** How frequently mouse information is sent to the server when a mouse button is down or up. Default = 40 down, 150 up.
- **DPI:** Resolution of the X server on the GO-Global Server. Default = 75.
- **Data cache:** Size of the RAM cache for data storage. Default = 16 MB.
- **Cache compression:** Whether data to be cached is to be compressed first. Default = None.
- **Keytable path:** The absolute path to the directory containing the region- and language-specific keyboard keytable files. Default = \$GOGLOBAL_ROOT/etc/keytables.
- **Extra server options:** Additional server-side arguments that configure performance or other options.

Note: With **Adaptive Tuning** enabled, some reported properties are tuned when the session is started, so they may not match the default values displayed on this screen.

Launching Applications from a Shortcut

GO-Global allows users to create shortcuts for launching applications without having to open the Program Window. For users running the native Windows Client, there are two methods for creating a desktop shortcut.

To create a desktop shortcut

1. Install the native Windows Client on the user desktop.
2. Once you are connected to the GO-Global Server, simply drag the application icon from the Program Window on to the user desktop.
3. Double-click the desktop shortcut to launch the application.

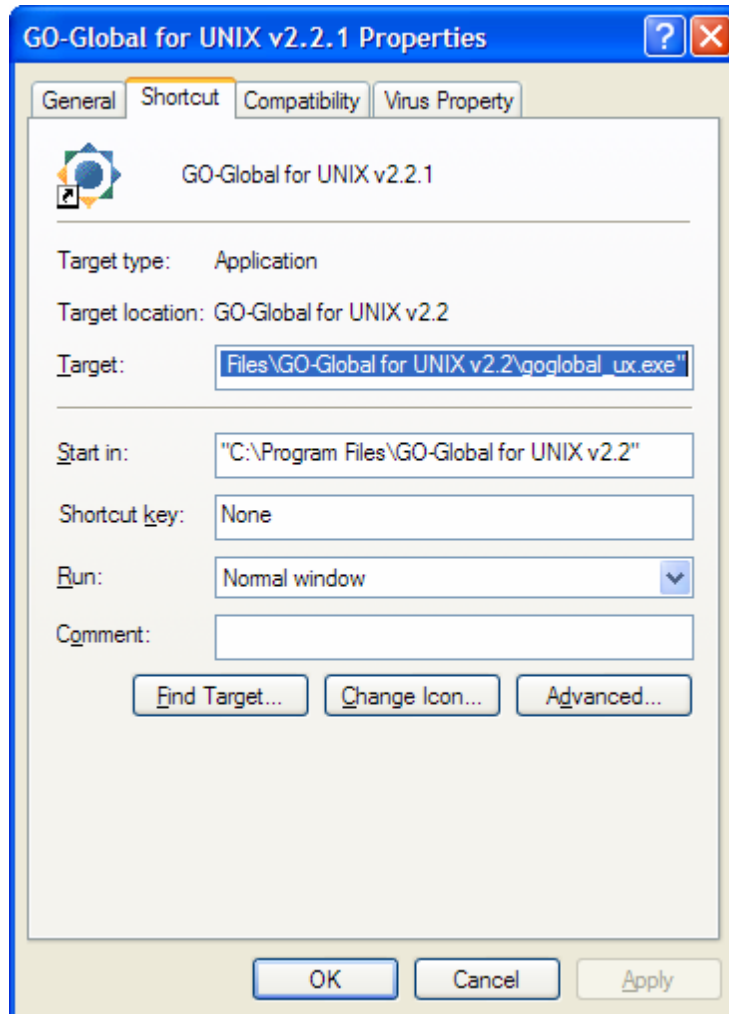
-or-

1. Install the native Windows Client on the user desktop.
2. Edit the desktop shortcut by highlighting the shortcut, right-click and select the **Properties** option.
3. Look at the existing command line in the **Target** field.
4. Edit the command-line to include the following arguments:

username	Client's network username
host	GO-Global Server IP address or hostname
transport	Transport protocol, typically 'tcp' or 'ssl'
launch	Display name of the application. The display name must match the entry in GO-Configure.

For example, if you want to launch xterm on a system called server, the command-line would appear as follows:

```
"C:\Program Files\GO-Global for UNIX v2.3\goglobal_ux.exe" username=joe
host=192.168.10.11 transport=tcp launch=Xterm
```



5. Click **OK** and double-click the desktop shortcut to launch the application

Note: See Chapter VI for additional command-line arguments.

Configuring the ActiveX Control

GO-Global provides web-based clients that can be accessed from the client machine's web browser. Depending on their choice of browser, users can connect to a GO-Global Server via an ActiveX control or a Mozilla plug-in.

GO-Global's ActiveX Control is available to users running Internet Explorer on a supported Windows client platform. Mozilla plug-ins are available to Windows-based clients only.

To configure the ActiveX Control

1. On the GO-Global Server or your web server, locate the directory that contains the GO-Global html files. The default directory is `$(GOGLOBAL_ROOT)/htdocs`. (If you're using Apache, the default document directory will likely be found in `/var/www/html`.)
2. If you plan to use the Active X Control function, locate the file **IPviewer.html**.
3. Edit the html file with vi or your favorite text editor.
4. Locate the section that starts with `<PARAM NAME=` and edit the corresponding values to reflect your hostname, application, and startup arguments.
5. Save the file as *application.html*.
6. From the user desktop, point the browser to <http://server/application.html>.
7. Allow the ActiveX Control to be installed. Use the mouse to drag the URL path to the desktop to create a shortcut.
8. Double-click the shortcut to launch the application.

Session Configuration

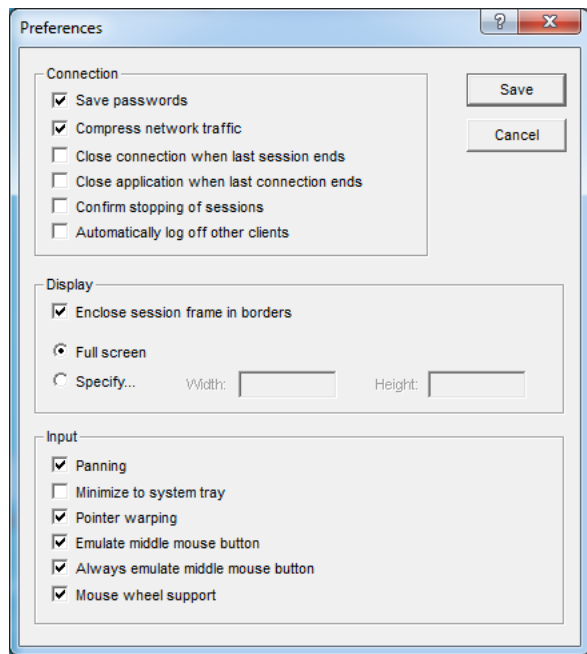
The appearance and performance of your GO-Global sessions can be configured to best suit your needs. GO-Global provides three modes of configuration, as follows:

- Published application configurations set using GO-Configure and stored on the host machine for distribution.
- Preferences settings on the client that set options.
- Command-line options configured and called when the application is launched from a command- line prompt or a shortcut.

For example, if the server database specifies that a GO-Global session be published in full screen mode but the Client Preferences settings specify width and height, the session will display in a window of the specified dimensions. If a command-line option specifies full screen mode thereafter, then Full Screen mode will be used.

Preferences

To open the Preferences dialog box, click Connection | Preferences.



Note: Preference settings do not take effect until a new connection is opened. Settings are shown above with their default values.

Connection

- **Save passwords**
By default, the usernames and passwords used to make connections are not retained on the client machine. To speed connections, enable **Save Passwords**. Upon selecting a server, the connection will be made automatically. Default = Off.
- **Compress network traffic**
Compression reduces the bandwidth required by the GO-Global connection. Under most circumstances performance will be better with compression enabled. Default = On.
- **Close connection when last session ends**
When the last running session ends, the client will disconnect from the host.
- **Close application when last connection ends**
When the last connection disconnects, the application will exit.
- **Confirm stopping of sessions**
When enabled, a confirmation dialog is presented when the user selects the "Stop" option when closing a session, connection, or client. Users can then confirm if they wish to stop all of the sessions listed, or cancel the operation.
- **Automatically log off other users (Windows Clients only)**
When enabled, gold will automatically suspend any sessions and log off any session leaders with running sessions. Sessions that were previously running on these leaders may be resumed on the newly connected client.

Display

- **Enclose session frame in borders**
Borders on smaller windows make moving and resizing the application more intuitive for some users, as well as provide access to menus and buttons. Full-screen windows without borders resemble a desktop session, providing the illusion of working in an X display environment. Default = On.
- **Full screen**
Display preferences can toggle the GO-Global windows between full screen and specified dimensions of Width and Height. Maximum size: 4096x2048. Default = Full Screen (single-window session only).

Input

- **Panning** (Windows Clients only)
When an X application window is partially off screen, moving the mouse to the edge of the screen will “jump” the window fully on screen.
- **Minimize to system tray** (Windows Clients only)
When the Connection Manager window is minimized, it will become an icon in the Windows System Tray — the area near the clock on the Windows taskbar.
- **Pointer warping**
When enabled, pointer warping allows an application to place the mouse cursor wherever it wants.
- **Emulate middle mouse button** (Windows Clients only)
Enables middle mouse emulation, meaning holding down the right and left mouse buttons at the same time is treated as if the user were pressing the middle mouse button.
- **Always emulate middle mouse button** (Windows Clients only)
When enabled, middle mouse emulation is not disabled when a real middle mouse button is pressed.
- **Mouse wheel support** (Windows Clients only)
When enabled, the mouse wheel is supported.

Command-Line Options

Command-line configurations override those published with the application on the server and those set in the client preferences. Command-line options can be called in the following places:

- The command-line or Windows **Run** prompt on the client, appended to the command path and name.
- The **Target** field of a Windows shortcut, edited by right-clicking the **GO-Global for UNIX v2.3.0** icon and selecting **Properties**.

The following options can be appended to a **goglobalux** command:

Command-Line Option	Function
<code>host={hostname IP_address}</code>	Presets the hostname or IP address of the GO-Global Server. Default value is the hostname or IP address of the host from which the client software was downloaded. If HOST is not specified, PORT, USERNAME, and PASSWORD are ignored.
<code>port=port#</code>	Presets the port number to be used to connect to the server. Default value is 491.
<code>username=username</code>	Presets user's login name. No default value. If not specified, user will be prompted.
<code>password=password</code>	Presets user's login password. No default value. If not specified, user will be prompted.

Command-Line Option	Function
<code>launch="Display Name"</code>	Causes the published item with the given <i>Display Name</i> to be started automatically after the connection is established. Note that the <i>Display Name</i> is case-sensitive and must exactly match the name of the published application. No default value.
<code>ui={min max tray hide}</code>	Controls how the connection manager user interface is initially displayed to the user. min = window is presented minimized; max = window is presented maximized; tray = only a tray icon is presented; hide = no user interface is presented. This may cause problems since the user will have no way to exit the application. No default value. The default display is normal window. Windows Client only.
<code>width=width</code>	Sets the maximum width, in pixels, of a single-window session window. No default value.
<code>height=height</code>	Sets the maximum height, in pixels, of a single-window session window. No default value.
<code>compress={true false}</code>	Enables or disables compression for data transmission between the server and client. On low-bandwidth machines, enabling compression increases session performance, but high-bandwidth connections may actually experience reduced speed, due to increased processing time. Default value is true (compression enabled).
<code>transport={tcp ssl httpsproxy}</code>	Transport protocol to use for data transmission between the server and client. Default transport is TCP/IP. The https option is for the Windows Client only.
<code>closeconnection={true false}</code>	When the last running session ends, the client will disconnect from the host.
<code>closeapp={true false}</code>	When the last connection disconnects, the application will exit.
<code>joinsession={true false}</code>	Users can opt to <i>not</i> join a currently running client instance with compatible attributes. When <code>joinsession=false</code> , GO-Global will cause the next invocation of the client to ignore any running Windows client instances that it might otherwise join. By default, users join a compatible instance of the client.

Table 6.1

Example:

The following command line launches an X terminal (on this server, it has the name "Xterm") for username demo on server myserver. SSL will be used as the transport, and the connection manager window will be minimized to the system tray. The user will be prompted for the login password.

```
"C:\Program Files\GO-Global for UNIX v2.3\goglobal_ux.exe" host=myserver
username=demo launch=Xterm ui=tray transport=ssl
```

Working in the XDM Environment

The X Display Manager (XDM) environment option allows the GO-Global client user to log in using the X Display Manager Control Protocol (XDMCP), offering the client user greater latitude in choosing the server serving the client. When a site has more than a few GO-Global Servers, XDMCP provides a way to manage resources. Since several variables are involved, you may want to experiment with various settings. XDM sessions are configured with GO-Configure.

XDM Servers must support reverse DNS to locate remote hosts.

Notes: When using XDM on a host with multiple network interfaces, add **-from \$HOSTNAME** to the **Extra Server Options** on the **Optimizations** tab. Not doing so can cause clipboard anomalies.

The following three XDM options are available and can be activated via GO-Configure:

<code>direct:hostname</code>	Directs an XDM request to <i>hostname</i> .
<code>indirect:hostname</code>	Provides a list of all servers available to host a session, from which the client may then choose one.
<code>broadcast</code>	Searches for all available servers and initiates an X session, then logs on to the first server that responds.

Table 6.2

In **direct** mode, the X server connects to an XDM host and requests the display management services of that host directly. This usually results in a username/password dialog.

In **indirect** mode, the X server connects to the XDMCP host and requests the display management services of a server, but not necessarily the immediately connected server. The normal way for XDM to handle this is to present a “chooser” list of available servers on the network, and allow the user to pick one. Once the user picks one, the X server reconnects to that XDM host and requests the display management services of that server, initiating a username/password dialog.

In **broadcast** mode, the X server broadcasts for the services of any server, and connects to the first available server that responds to the broadcast.

Note: Setting the **XDM Mode** to anything other than **None** will disable the **Command Line** field and the **Run in Own Session** checkbox on the **Description** tab, because XDM modes do not have a command line and must always run in their own session.

Caution! Do not set window size to less than 400x400 or the login dialogs will be too small to use.

Enhancing Session Performance

The following steps may enhance session performance, depending on the connection, the platform operating system, and the published applications:

1. The **Server Back Buffer** option reduces the amount of data the X servers require from the client. If frequently used X applications are dependent on client bitmaps, setting the **Server Back Buffer** to **On** will improve performance.
2. On display devices running 16-bit color or higher, better performance can be achieved by disabling **Multidepth** support. **Multidepth** has no effect if the display device is in 8-bit color mode, because only an 8-bit visual is supported in this case. **Multidepth** support provides an 8-bit default visual for compatibility with X applications that do not work well with a visual that is deeper than 8 bits. However, on low-bandwidth connections, the 8-bit default will perform better, because 8-bit images require less bandwidth than images greater than 8 bits deep.
3. To conserve bandwidth when working over slower connections, try the following:
 - Reduce the window size of the session or application.
 - Reduce the display color depth of the application via GO-Configure.
 - Turn off the blinking cursor in terminal windows (e.g., `dtterm` and `xterm`).
 - Use a solid-colored background for remote desktop displays.

Enhancing Client Security

Following are some of the steps that can be taken to maximize security to protect data, whether it is on the server, the client, or in transmission.

- Use SSL instead of TCP/IP for your **Transport** preference. It provides at least 56-bit encryption, whereas TCP/IP transmits all data in unencrypted format. For maximum security, GraphOn also offers an enhanced encryption installation that allows 256-bit AES encryption.
- Do not enable the **Save Passwords** option under Preferences.
- When using a browser to connect to the GO-Global Server, always purge the browser's cache after closing your connection.
- Avoid using desktop icons to quickly access remote applications, as they will be immediately visible and may provide account information. Review the shortcut's properties to determine what information may be available.

CHAPTER VII – Troubleshooting & Technical Support

7

Log Files

The following logs and message files are provided as diagnostic tools. They can be useful either for troubleshooting or for providing additional information to customer support if such assistance is necessary.

Installation Logs

If you have a problem with the server installation, please look in: `/tmp/graphon_installer_logs/`.

gdbi.log

During installation, the GO-Global database is initialized. The output of the program that initializes the database is saved as `/tmp/graphon_installer_logs/gdbi.log`.

Master Log

The **master** gold process accepts incoming connections and authenticates users. It is also responsible for certain maintenance procedures such as expiring suspended sessions.

The log file for these activities is configured using the `$(GOGLOBAL_ROOT)/etc/masterlog.conf` file. By default, the file `$(GOGLOBAL_ROOT)/etc/gold-$(HOSTNAME)-master.log` is used and is rotated when the file size is greater than 10 megabytes (up to a maximum of 10 logs). The `masterlog.conf` file can be used to adjust these settings as well as specific levels of logging. Changes to the `masterlog.conf` will only take effect if the master gold process is stopped and then restarted.

Leader Log

The **leader gold** process is started after a user has successfully authenticated from a client. Each client connection will have a leader log specific to that particular connection. This process will publish the configured applications to the user as well as start/suspend/resume/stop sessions as directed.

The log file for these activities is configured using the `$(GOGLOBAL_ROOT)/etc/leaderlog.conf` file or `$(HOME)/.graphon/leaderlog.conf` if it exists. By default, the file `$(HOME)/.graphon/log/$(USER)-$(HOSTNAME)-leader.log` is used and is rotated whenever a new leader process is started. The leader logs are kept up to a certain number of days (14 by default). The `leaderlog.conf` file can be used to adjust these settings as well as specific levels of logging. Changes to the `leaderlog.conf` will take effect the next time a leader gold process is started.

Session Log

A **session** corresponds to the startup and running of a specific instance of the GO-Global X server. This will include the running of the X server, Window Manager, and X applications.

The log file for these activities is configured using the `$(GOGLOBAL_ROOT)/etc/sessionlog.conf` file or `$(HOME)/.graphon/sessionlog.conf` if it exists. By default, the file `$(HOME)/.graphon/log/$(USER)-$(DISPLAY)-session.log` is used and is rotated whenever a new session is started. The session logs are kept up to a certain number of days (14 by default). The `sessionlog.conf` file can be used to adjust these settings. Changes to the `sessionlog.conf` will take effect the next time a session is started.

syslog

The GO-Global Login Daemon (**gold**) reports error conditions to the UNIX **syslog** daemon via the USER facility. **syslog** will deal with those error reports according to its configuration file — typically `/etc/syslogd.conf`.

Xerrors

If you experience problems running **xdm** on an HP-UX host, check the file `/var/dt/Xerrors` for any messages regarding fonts. If you find any such messages, verify that the X font server (**xfs**) is running on the host. If it is not, follow these steps to start it:

1. Edit the file `/etc/rc.config.d/xfs` and set the variable `RUN_X_FONT_SERVER=1`
2. Execute the command `# /sbin/init.d/xfs start`
3. Add the font server to your font path.

Client Problem Situations

Cannot connect to server

1. Verify your username and password.
2. Verify that you can connect to the GO-Global host through other means (for example, a web browser or the `ping` program).
3. If you are using a nondefault port number, verify that you have entered both the server name and the port number separated by a colon. For example, `ourserver:492`
4. Contact your network administrator and ensure that the GO-Global host is up and that the GO-Global Server software is running.

Note: When the GO-Global License Manager Daemon (lmgrd) is unreachable, GO-Global Login Daemon (gold) exits and users' sessions close.

Cannot launch a session

1. Verify that you are still connected to the host. (It will be listed on the **Connections** menu under **Current Connections**.)
2. If you receive the error message, *License unavailable*, contact your system administrator.
3. Check your `$HOME/.graphon/session-logs` directory on the GO-Global host for any error messages.

Cannot publish, edit, or delete applications using GO-Configure

1. If the **GO-Configure** icon does not appear in your GO-Global client's list of published applications, your system administrator has not given you permission to use it. If you think you should have permission, contact your system administrator.
2. Editing and deleting: Verify that the application is one you published. You cannot edit or delete globally published applications.
3. Publishing: Press **F5** in the Windows Client to refresh your list of published applications, and verify that the application you are trying to publish is not already there.
4. Contact your system administrator and verify that GO-Global is properly configured.

License Management Logging

The GO-Global license management subsystem logs messages to several locations. General log information is logged to the master log file (but these messages are not described in the section on gold logging). Additional error information is logged to the system logs (usually in `/var/adm/messages` or `/var/log/syslog/syslog.log`).

Messages logged by the license management subsystem to the gold log file are:

- `Unable to open licensing library`
Gold was unable to open the `libgolicense.so` shared object library that contains the code for the license management subsystem. The reason for the failure, as reported by the operating system, is also logged. The failure reason as returned by the operating system should suggest a course of action to resolve this problem.
- `Unable to open licensing library (corrupt library?)`
This message indicates that gold found the licensing library but could not locate the shared objects it is required to contain. This indicates that the library has been corrupted or tampered with. The solution to this problem is to replace the library with a known valid copy. Additional diagnostic information as returned by the operating system is also logged with this message.

The individual licensing shared object libraries also log diagnostic information. For technical reasons, these messages are not logged to the master log file, but instead are logged to the system error log (usually `/var/adm/messages` or `/var/log/syslog/syslog.log`).

The messages logged by the FlexNet Publisher license management library (`libflexlmlicense.so`) are:

- `GO-Global UX license allocation failed: [error]`
This indicates that FlexNet Publisher was unable to check out a license as requested. The reason for the failure is indicated by the error returned from FlexNet Publisher. Please consult the FlexNet Publisher reference guides for further information regarding specific errors encountered.

- GO-Global UX License failure: [error]

This indicates that FlexNet Publisher was unable to check out a license as requested. The reason for failure is indicated by the error returned from FlexNet Publisher. Please consult the FlexNet Publisher reference guides for further information regarding specific errors encountered.

- GO-Global UX License failure: license revoked

This indicates that your license has been revoked by GraphOn. Please contact your GraphOn sales representative to resolve this situation.

At startup, GO-Global attempts to obtain a license to enable the enhanced 168-bit encryption by attempting to check out a license for the “UX_encryption” feature. If you have not purchased this option, a message is logged to the system log to the effect that the license check-out failed. This is normal and is not cause for alarm.

FlexNet Publisher License Error Codes

The following list includes the most commonly seen FlexNet Publisher error codes and descriptions of the situations they were seen in. For detailed information regarding FlexNet error codes, refer to the *License Administration Guide*, available at http://www.globes.com/support/fnp_utilities_download.htm#downloads

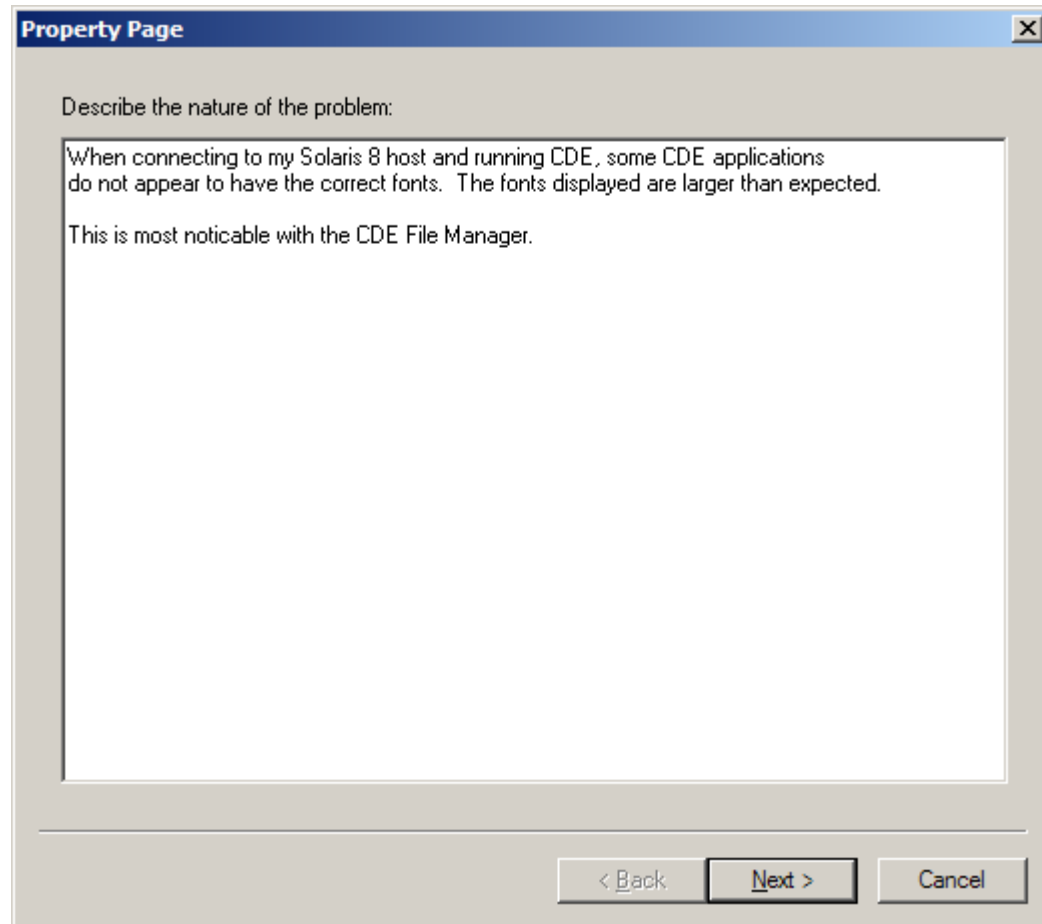
- FLEXlm error: -2,413.

“Invalid license file syntax.” The host ID specified in the license file does not match the server’s host ID.

Getting Help (Reporting a Problem)

The GO-Global client offers a way to create a “problem report”, including relevant log files, which can be submitted to GraphOn. To use this feature, select **Report a Problem** from the **Help** menu. This will start a wizard that will help you collect the relevant information for your problem.

The first dialog in the wizard will ask for a description of the problem.



Property Page

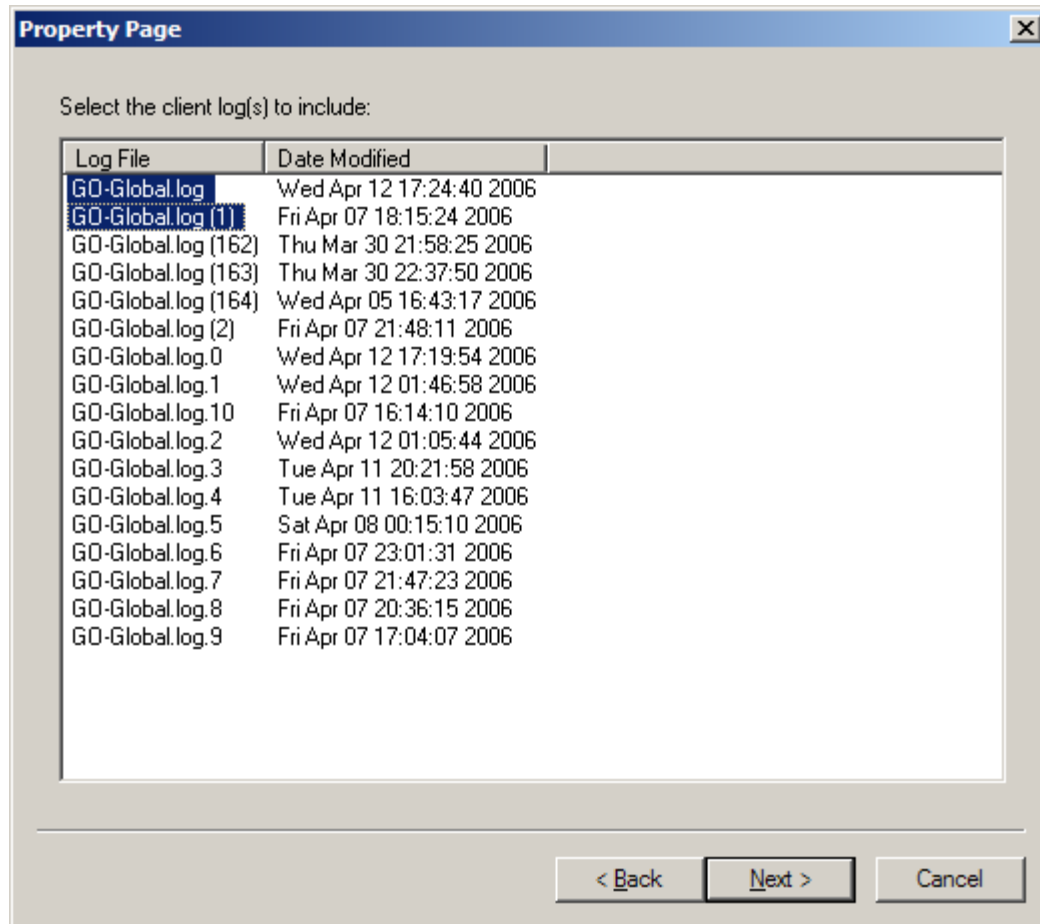
Describe the nature of the problem:

When connecting to my Solaris 8 host and running CDE, some CDE applications do not appear to have the correct fonts. The fonts displayed are larger than expected.

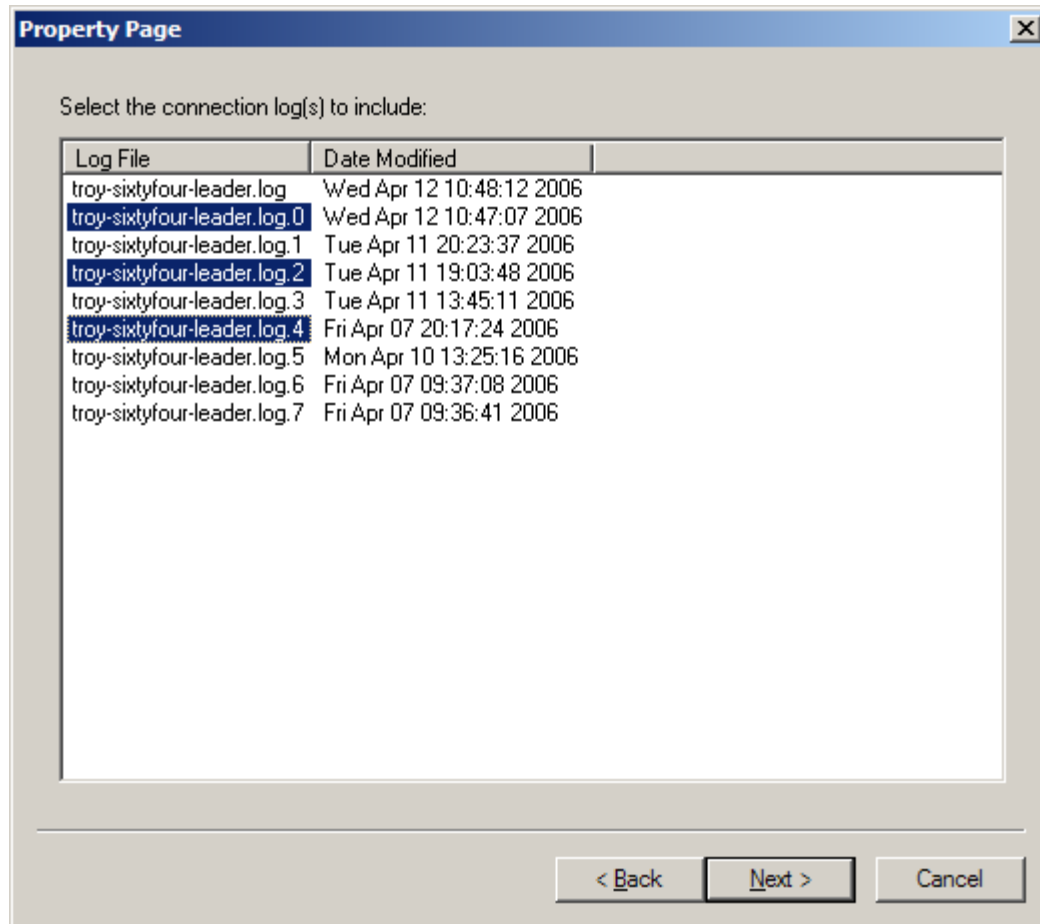
This is most noticeable with the CDE File Manager.

< Back Next > Cancel

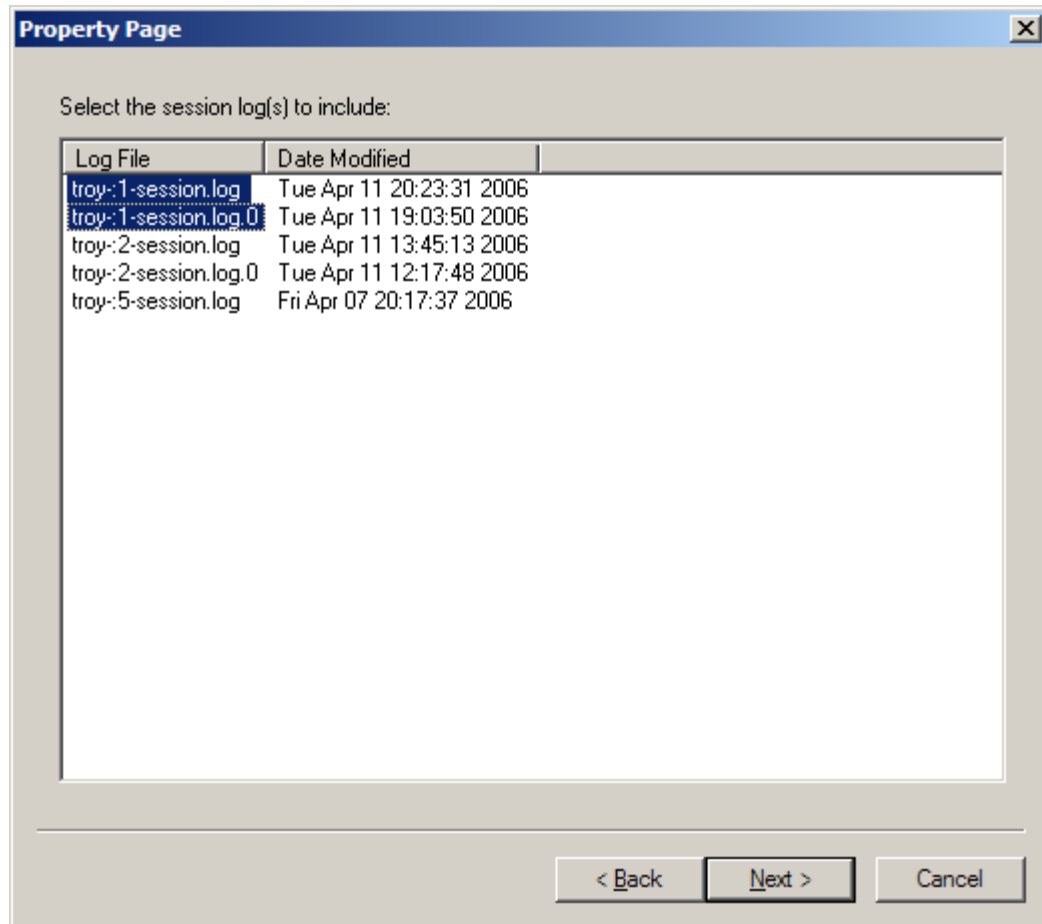
Clicking the **Next** button will take you to the client log selection page. On this page, all available client logs and their timestamps are listed to help identify the most recent. You can select any or all of these log files to include in your problem report.



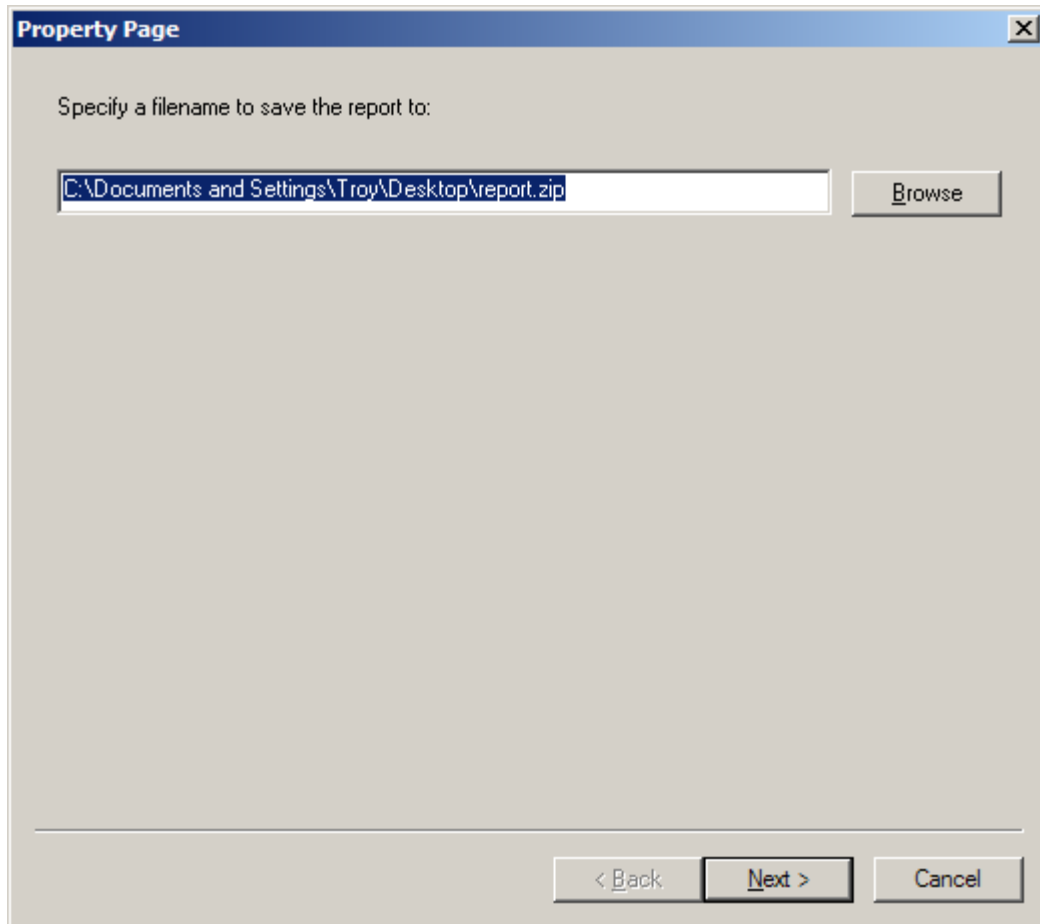
Clicking **Next** will cause the wizard to display the connection log selection page. This page lists all connection-level logs as they exist on the server. You can select any of all of these logs to include with your problem report. Note that only logs from the host of the currently active connection tab are listed.



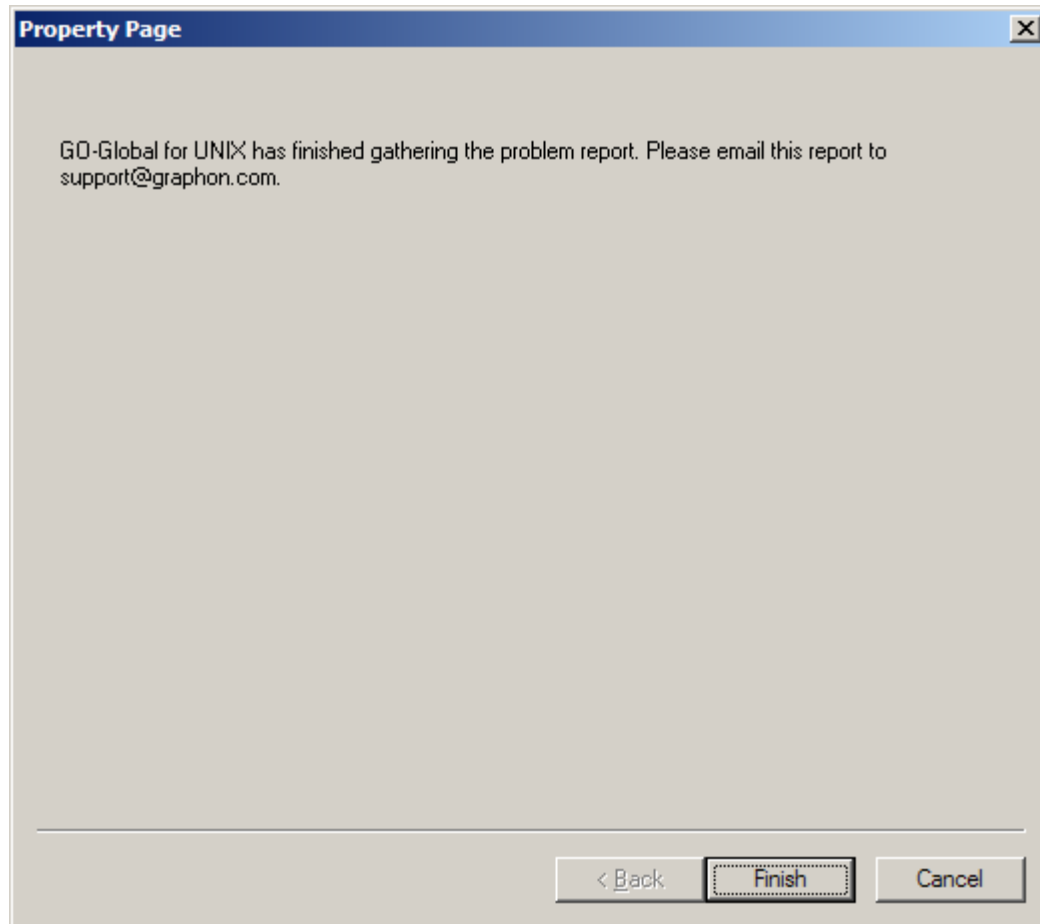
Clicking **Next** will display the session log selection page. From here, select the session log(s) that you want to include with your problem report.



Clicking **Next** will cause the wizard to prompt for a filename to save the report into. Select a suitable location and filename for your problem report.



Finally, the wizard will create the problem report and ask that it be sent to GraphOn.



Getting Help (Technical Support)

See our web site at www.graphon.com/support for the most up-to-date information on all GO-Global support issues, including:

- The latest documentation
- Software updates
- Frequently asked questions (FAQ)
- Workarounds for known problems
- Technical notes
- Product news

GraphOn offers a variety of technical support and maintenance options, including an informative web site and assistance by e-mail or telephone. For information on purchasing technical support and maintenance contracts, e-mail sales@graphon.com.

To contact GraphOn Technical Support, e-mail support@graphon.com. All support requests should be accompanied by error messages and system logs relevant to your problem. Please note that using the support form located at www.graphon.com/support will often lead to the quickest response from the support team. We also encourage you to visit <http://www.graphon.com/support/forum/> to join an online forum of GO-Global users that posts Frequently Asked Questions, feature requests, and general support issues.

Certificates

At installation time, the GO-Global installation program creates a default server key certificate and a default *Certificate Authority* (CA) certificate to sign the server key. These certificates are created with a default lifetime of five years.

Certificate Authority

The installer-created CA certificate has these properties:

- Country: US
- State: Unknown State
- Locality: Unknown City
- Organization: GraphOn GO-Global for UNIX User
- Organizational Unit: Unknown Department
- Common Name: GraphOn GO-Global for UNIX Default CA
- E-mail Address: `root@hostname`

The hostname in the **E-mail Address** field is that of the host on which GO-Global for UNIX is being installed.

The CA certificate is self-signed and will appear to be issued by “GraphOn GO-Global for UNIX Default CA” to “GraphOn GO-Global for UNIX Default CA”.

The server key and CA certificates are placed in the `$GOGLOBAL_ROOT/ssl/certs` directory as the files `CA.key` and `CA.crt`, respectively.

Server Key Certificate

The installer will use the default CA certificate to generate a signed server key certificate with the following properties:

- Country: US
- State: (two-letter abbreviation)
- Locality: Unknown City
- Organization: GraphOn GO-Global Customer
- Organizational Unit: Server Key
- Common Name: *hostname*
- E-mail Address: *root@hostname*

The hostname in the **Common Name** and **E-mail Address** fields is that of the host on which GO-Global is being installed. The server key certificate will appear to be issued to whatever the hostname is, by “GraphOn GO-Global for UNIX Default CA”.

Obtaining a Trusted Server Certificate

GO-Global for UNIX can use any server certificate to provide increased security. To obtain a server certificate from a CA trusted by the client operating system, consult the documentation from the CA of your choice using the following information as a guide.

The CA will require a *Certificate Signing Request* (CSR) from you. Generally this can be generated by running:

```
# $GOGLOBAL_ROOT/bin/openssl req -new -key server.key  
-out server.csr
```

Running this command will prompt you for the attributes to be included in your certificate, as follows:

- Country Name: US
- State: your state
- Locality: your city
- Organization: your company name
- Organizational Unit: your department
- Common Name: your server’s name
- E-mail Address: your e-mail address

It is essential that the **Common Name** matches the hostname of your GO-Global Server. Any variation in the name will cause the client to issue a warning when connecting.

The output of the above command will be a file named `server.csr`, which can be sent to your CA. When your CA sends you the certificate, place it in the file `$GOGLOBAL_ROOT/ssl/certs/server.crt`.

Since GO-Global’s SSL implementation is based on the OpenSSL toolkit, the tools used are the same as those used in other OpenSSL-based products, such as the Apache **mod_ssl** package. You may be able to follow instructions provided by your CA for the **mod_ssl** package to obtain a GO-Global certificate. GO-Global includes the **openssl** command-line tool in the `$GOGLOBAL_ROOT/bin` directory. This tool performs all key and certificate functions in the OpenSSL and OpenSSL-derived products.

Creating a Site Certificate Authority

Sites with many GO-Global Servers installed may wish to create their own certificate authority, then sign each server's certificate from this authority, and then install the certificate authority certificates onto each client. This will prevent any "warnings" about untrusted authorities, without requiring the site to obtain a third-party certificate for each server.

Note that there are many third-party applications and systems to assist in the creation and maintenance of a certificate authority that interoperate with the OpenSSL toolkit. These tools should be able to generate signed server certificates for use with GO-Global without modification. In addition, a site may opt to have a server key signed by an already established Internet certificate authority, such as Verisign or Thawte. However, as shown below, it is possible to create and manage your own site-wide certificate authority using only the tools provided with GO-Global for UNIX.

Preliminary Considerations

Like any cryptographically secure application, the OpenSSL toolkit requires a source of entropy, or random numbers. Preferably this source should *not* be from a pseudorandom number generator, but should instead be based on system events, such as the timings between keystrokes, I/O events, and so on.

Many recent operating systems include devices in the `/dev` filesystem for this purpose. The `/dev/random` device outputs cryptographically strong random numbers from a fixed pool of entropy. Once the entropy in the pool is exhausted, `/dev/random` will block until more entropy is made available. The `/dev/urandom` device outputs cryptographically strong random numbers from the same entropy pool, but unlike `/dev/random`, `/dev/urandom` will continue to output random numbers based on a pseudorandom number generator if the entropy pool is exhausted. Therefore, `/dev/urandom` is often more suitable for server applications (or daemons) that cannot be blocked if the entropy pool is exhausted.

Official patches exist to provide `/dev/random` and `/dev/urandom` on Solaris systems. GraphOn highly recommends the use of these patches.

If you do not have the `/dev/random` or `/dev/urandom` devices, you can simulate some pseudo-random data by creating a file that contains snapshot information about the system, such as:

```
(ps; date; who; last | head -30) > randomdata
```

In any event, you should set the `RANDFILE` variable to point to your entropy source, such as one of:

```
export RANDFILE=./randomdata
export RANDFILE=/dev/urandom
```

GraphOn recommends allocating a specific directory to your certificate authority (for example, `${GOGLOBAL_ROOT}/CA/`), and performing all certificate authority operations from this location.

Establishing the Certificate Authority

A certificate authority is a virtual organization that will sign each of your server keys, allowing the client to assert that the server keys are authentic and have not been tampered with.

To establish the certificate authority, a CA key and self-signed certificate are created. Once the CA certificate and key are created, the CA certificate is imported into the Internet options on the client PC. Finally, the server keys are signed using the CA certificate, which will allow the client PCs to recognize the authenticity of the signatures and allow connections to the server without warning the user about the trustworthiness of the CA.

Creating a CA Key and Certificate

The certificate authority's key is its private key in terms of the RSA technology. This key should be kept very secret, as any entity with access to this key can generate false certificates that would certify unknown hosts as trusted. It is vitally important to protect the integrity of your certificate authority.

To generate the CA key, use the following command (the command appears split on to two lines here but should be typed as one continuous command-line):

```
# OPENSSL_CONF=${GOGLOBAL_ROOT}/ssl/openssl.cnf
${GOGLOBAL_ROOT}/bin/openssl genrsa -out ./ca.key 1024
```

This command will generate your initial CA key, and place it in the file "ca.key". After the key is created, we must generate a Certificate Signing Request (CSR) that we will use to create the CA certificate. We do this using the following command:

```
#OPENSSL_CONF=${GOGLOBAL_ROOT}/ssl/openssl.cnf
${GOGLOBAL_ROOT}/bin/openssl req -new -key ./ca.key -out ./ca.csr
```

This command will run interactively and prompt you for the information to be contained in the certificate. Example responses are shown here in *italics*:

```
Using configuration from /usr/local/graphon/ssl/openssl.cnf
You are about to be asked to enter information that will be
incorporated into your certificate request.
What you are about to enter is what is called a Distinguished Name
or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:US
State or Province Name (full name) [Some-State]:Washington
Locality Name (eg, city) []:Bellevue
Organization Name (eg, company) [Internet Widgits Pty Ltd]:GraphOn
Corporation
Organizational Unit Name (eg, section) []:GraphOn Corporation CA
Common Name (eg, YOUR name) []:GraphOn Corporation CA
Email Address []:hostmaster@graphon.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:[enter]
An optional company name []:[enter]
```

Specifically, the prompts should be answered as:

- **Country Name:** your two-letter country abbreviation
- **State or Province Name:** your full state or province name
- **Locality Name:** your city or town or suburb name
- **Organization Name:** the name of your organization or company
- **Organizational Unit Name:** the organizational name should be a representation of your CA's name
- **Common Name:** This should either be a person responsible for the operation of the CA or a generic name representing the CA itself

- **Email Address:** This should be an e-mail address that can be used to address concerns about certificates to someone responsible for the CA

The final step is establishing the CA certificate. To do this, you must create a settings file that contains some information about the CA. The file should be named **ca.cfg** and should contain the following:

```
extensions = x509v3
[ x509v3 ]
subjectAltName = email:copy
basicConstraints = CA:true,pathlen:0
nsComment = "[your company] site CA"
nsCertType = sslCA
```

After creating this file, you can sign your CA certificate with the following commands:

```
# OPENSSL_CONF=${GOGLOBAL_ROOT}/ssl/openssl.cnf
${GOGLOBAL_ROOT}/bin/openssl x509 -req -extfile ./ca.cfg -days
1825 -signkey ./ca.key -in ./ca.csr -out ./ca.crt
```

The resulting certificate file, **ca.crt**, is the certificate that will need to be imported into the certificate store on each client PC.

It is also necessary to create a configuration file for signing server keys. This file should be named **server.cfg**, and should contain:

```
extensions = x509v3
[ x509v3 ]
subjectAltName = email:copy
nsComment = "Certificate signed by your company CA"
nsCertType = server
```

You must also create a file that will store the serial numbers of certificates signed by this CA. Use the following command:

```
# echo "01" > ca.serial
```

Creating and Signing Server Keys

The default GO-Global installation will create a new certificate authority key and a new server key, signed by the newly created certificate authority. If you are trying to make sure all of your server keys are signed by your own certificate authority, you can begin by using the already created server key (in `${GOGLOBAL_ROOT}/ssl/certs/server.key`) or you can create your own.

To create a new server key, use the following command:

```
# OPENSSL_CONF=${GOGLOBAL_ROOT}/ssl/openssl.cnf
${GOGLOBAL_ROOT}/bin/openssl genrsa -out ./server.key 1024
```

This will generate a new server key and place it in the file `server.key`. Next, generate a Certificate Signing Request (CSR) for the server key. This is essentially the same process used for generating the CSR for the CA key, but the inputs are slightly different. Use the following command:

```
# OPENSSL_CONF=${GOGLOBAL_ROOT}/ssl/openssl.cnf
${GOGLOBAL_ROOT}/bin/openssl req -new -key ./server.key -out
./server.csr
```

This command will run interactively and prompt you for information about the server certificate that will be generated. Example input is shown here in italics:

```
Using configuration from /usr/local/graphon/ssl/openssl.cnf
You are about to be asked to enter information that will be
incorporated into your certificate request.
What you are about to enter is what is called a Distinguished Name
or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:US
State or Province Name (full name) [Some-State]:Washington
Locality Name (eg, city) []:Bellevue
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Company
Name
Organizational Unit Name (eg, section) []: GO-Global UX Server
Common Name (eg, YOUR name) []:dreadnought.sea.graphon.com
Email Address []:user@company.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []: [enter]
An optional company name []: [enter]
```

Your answers to these prompts should be:

- **Country Name:** Your 2-letter country abbreviation
- **State or Province Name:** Your full state or province name
- **Locality Name:** The city, town, or suburb where your organization is located
- **Organization Name:** The name of your company or organization
- **Organizational Unit Name:** Either a department name or some name representing this server
- **Common Name:** The name of this server, as it should appear on the certificate. Note that this is *not* the name of a person.
- **Email address:** The e-mail address of a party responsible for this server

Note that the **Common Name** entry is vitally important. It must *exactly* match the name of the server that the client will be connecting to. Generally, the common name is entered as a fully-qualified host name (i.e., **server.my.domain.com** instead of just **server**).

Finally, sign the server's key with the CA's certificate. This can be done using the following command:

```
# OPENSSL_CONF=${GOGLOBAL_ROOT}/ssl/openssl.cnf
${GOGLOBAL_ROOT}/bin/openssl x509 -req -extfile ./server.cfg -days
1825 -CA ./ca.crt -CAkey ./ca.key -CAserial ./ca.serial -in
./server.csr -out ./server.crt
```

Note that the **-days 1825** parameter will cause our server certificates to expire in 1825 days, or roughly 5 years time. If you want certificates to expire earlier or later, adjust this number to fit your requirements.

Copy the `ca.crt`, `server.key` and `server.crt` files to the target server into the `${GOGLOBAL_ROOT}/ssl/certs/` directory.

Your GO-Global Server now has a new SSL certificate, signed by your own custom certificate authority.

Redundant License Servers

When end users' data is divided among multiple servers and work remains possible if one of these servers goes down or is off line, multiple license servers can be employed. Two implementation methods can be used: redundancy via a license file list in the `LM_LICENSE_FILE` environment variable, or establishing a set of two or three redundant license servers. The `LM_LICENSE_FILE` variable is defined in the `${GOGLOBAL_ROOT}/etc/goglobalux` startup script.

Two-Server Redundancy

Two-server redundancy is designed to provide hardware failure protection only and does not provide load-balancing. This is because with two-server redundancy, only one of the two servers is "master" and capable of issuing licenses. The directions below explain the configuration of two-way redundancy via a license file list in the `LM_LICENSE_FILE` environment variable.

1. Obtain two separate license files that are hosted for each license server.
2. Copy the new `license.dat` files into the `${GOGLOBAL_ROOT}/etc` directory on both hosts.
3. Modify `${GOGLOBAL_ROOT}/etc/goglobalux` on the GO-Global Server on both hosts as follows:
`GG_LM_LICENSE_FILE="@hosta.domain.com:@hostb.domain.com"`
4. Issue a `goglobalux` stop and then start on both license servers.

We recommend running Flexera's **lmstat** utility to check the status of the redundant license servers once both servers are up and running.

Run `${GOGLOBAL_ROOT}/bin/lmstat -a -c ${GOGLOBAL_ROOT}/etc/license.dat` and verify that your servers are "UP".

You can obtain Flexera's `lmutil` tools (which include `lmstat`, `lmhostid`, `lminstall`, `lmremove`, etc.) from the `bin` directory (`${GOGLOBAL_ROOT}/bin` directory) or from: http://www.globes.com/support/fnp_utilities_download.htm#downloads. These tools are included for diagnostic purposes. Any questions on its functionality should be directed to Flexera.

Three-Server Redundancy

With three-server redundancy, if any two of the three license servers are up and running, a "quorum" of servers is established, and the system is functional and serves its total complement of licenses.

Three-server redundancy is designed to provide hardware failure protection only and does not provide load-balancing. This is because with three-server redundancy, only one of the three servers is "**master**" and capable of issuing licenses.

The following requirements are necessary to use the redundant licenser mechanism:

- You must have three servers (no more and no less) to install the license and FlexNet Publisher software on.
- At least two of the three servers must be up and running at all times.

Following is an example of a three-server redundant license file that GraphOn supplies after registering online. You must provide the hostnames of the three GO-Global Servers as well as the hostids for each. The port of the license server (e.g., 27000) must also be appended to each server line, if it is not already listed.

27000 is a good port number to use, unless you have other Flexera license managers already using that port. If so, choose another available port (e.g., 27001, 27002, etc.)

```
SERVER wilson 001122334455 27000
SERVER piper 010101010101 27000
SERVER caspian 000c297bbdf7 27000
VENDOR blm
INCREMENT unix_session blm 2.3 permanent 5 38D4C15B6410 \
    ck=303 SN=8675309-AA
INCREMENT any_app blm 2.3 permanent uncounted 74B546D236CE HOSTID=ANY \
    ck=143 SN=8675309-AA
INCREMENT StrongEncryption blm 2.3 permanent uncounted 9DE682CC84F4 \
    ck=75 SN=8675309-AA
```

The three-server license file needs to be copied to each of the three license servers.

Lastly, you must point the GO-Global Server to the license server. This can be done in two different ways, either by copying the license to each GO-Global Server and editing it to use USE_SERVER (see example below), or by adding each server to the `LM_LICENSE_FILE` environment variable defined in `${GOGLOBAL_ROOT}/etc/goglobalux`.

```
SERVER wilson 001122334455 27000
SERVER piper 010101010101 27000
SERVER caspian 000c297bbdf7 27000
USE_SERVER
```

To use the second option, change the value of the `LM_LICENSE_FILE` variable in `${GOGLOBAL_ROOT}/etc/goglobalux` from:

```
LM_LICENSE_FILE=${GOGLOBAL_ROOT}/etc/license.dat
```

to refer to the servers, like:

```
LM_LICENSE_FILE=27000@wilson,27000@piper,27000@caspian
```

Then, as root on each server, restart the GO-Global Server:

```
# ${GOGLOBAL_ROOT}/etc/goglobalux stop
# ${GOGLOBAL_ROOT}/etc/goglobalux start
```

We recommend running Flexera's **lmstat** utility to check the status of the redundant license servers once all three servers are up and running. Run `${GOGLOBAL_ROOT}/bin/lmstat -a -c ${GOGLOBAL_ROOT}/etc/license.dat` and verify that your servers are "UP".

You can obtain Flexera's **lmutil** tools (which include **lmstat**, **lmhostid**, **lminstall**, **lmremove**, etc.) from the bin directory (`${GOGLOBAL_ROOT}/bin` directory) or from:

http://www.globes.com/support/fnp_utilities_download.htm#downloads. These tools are included for diagnostic purposes. Any questions on its functionality should be directed to Flexera.

License File Method

With license file redundancy, each of a group of license servers serves a subset of the total number of licenses. The `LM_LICENSE_FILE` environment variable on each license server is set to specify a list of license files, where each license file refers to one of the license servers. Client connection attempts then try each server in turn until they either succeed or get to the end of the list. The servers can be physically far apart (unlike the case of three-server redundancy).

For example, if 10 licenses were desired for both Feature f1 and Feature f2, GraphOn could issue two sets of licenses with a count of 5 for each feature. The `license.dat` files would look something like this:

`license.dat` for server **acadia**:

```
SERVER acadia 17007ea8 1700
DAEMON xyzd $GOGLOBAL_ROOT/bin
FEATURE f1 xyzd 1.000 01-jan-2014 5 26C7DD9C0186
FEATURE f2 xyzd 1.000 01-jan-2014 5 8CE46C57041D
```

`license.dat` for server **concord**:

```
SERVER concord 17007ea8 1700
DAEMON xyzd $GOGLOBAL_ROOT/bin
FEATURE f1 xyzd 1.000 01-jan-2014 5 16BE40E1D98D
FEATURE f2 xyzd 1.000 01-jan-2014 5 6DB6F3E402DF
```

Users in one area could set their `LM_LICENSE_FILE` environment variable to:

```
1700@acadia:1700@concord
```

Users in another area could set their `LM_LICENSE_FILE` environment variable to:

```
1700@concord:1700@acadia
```

Client connection attempts would first be made to the first server in the list. If that failed for any reason, the second server would be tried.

Mapping a Locale

To map a locale reported by the GO-Global client to a specific locale of your host, edit `$GOGLOBAL_ROOT/etc/localemap` using a text editor. Create a line like the following:

```
<lang>_<SUBLANG>    <locale name>
```

`$GOGLOBAL_ROOT/etc/localemap` contains a list that has the possible `<lang>_<SUBLANG>` locales reported by the GO-Global client. You may need to verify that your host has a valid `<locale name>` for each relevant entry. This can be done using **locale -a** to see the locales available on your host. In some cases, your host may have alternate `<locale name>` values you may wish to substitute. By default, an undefined `<lang>_<SUBLANG>` will use the **C** locale. After making any changes to the locale map, gold will need to be restarted, like this:

```
# $GOGLOBAL_ROOT/etc/goglobalux stop
# $GOGLOBAL_ROOT/etc/goglobalux start
```

Mapping a Time Zone

To map a time zone reported by the GO-Global client to a specific time zone for your host, edit `$GOGLOBAL_ROOT/etc/timezonemap` using a text editor. Create a line like the following:

```
<timezone>    <TZ_NAME>
```

`$GOGLOBAL_ROOT/etc/timezonemap` contains a list that has the possible `<timezone>` values reported by the GO-Global windows client. Most of the common values are defined already, but your host may desire different values (e.g., UTF-8 instead of PST-8PDT). After making any changes to the time zone map, gold will need to be restarted as follows:

```
# $GOGLOBAL_ROOT/etc/goglobalux stop
# $GOGLOBAL_ROOT/etc/goglobalux start
```

Support for Modular Authentication (PAM)

GO-Global's authentication uses the **Pluggable Authentication Modules** (PAM) architecture on supported hosts. On hosts that do not support PAM, native modular authentication architectures are used. Using a modular authentication system such as PAM allows the GO-Global Server to leverage the existing authentication modules on the system, making authentication through GO-Global more closely resemble authentication through other services (e.g. telnet, ssh, XDM, etc.). In most cases, if the host is already configured to use a particular authentication module for telnet or login access (e.g., authenticating against an LDAP database, or authenticating using cardkeys or crypto-cards), the same modules can be used with GO-Global.

Modular authentication systems use a *conversation* between the authentication API and the user. This allows the user to interact with the authentication modules, and to answer additional authentication questions or satisfy additional authentication requirements. In the GO-Global implementation, questions from the PAM (or other authentication systems) API are remoted to the thin client where the user is presented with the query or information. Responses collected are transmitted back to the host for submission to the API. Following are two example conversations:

```
Username: joe
Password: foo
Your passwords has expired
New password: bar
Re-enter new password bar
Password change accepted
Authentication Succeeded
```

```
Login: joe
Challenge #: 356268234
SecurID #: 463939
Invalid SecurID value
Challenge #: 233895723
SecurID#: 633354
Authentication Succeeded
```

Each application that uses the PAM system for authentication may have its own individual configuration profile. This profile dictates which modules should be used for authentication for that particular application. GO-Global will use a profile called `goglobal` if it exists, otherwise it will use the default `other` profile. To configure your GO-Global server to use PAM to authenticate, you must create a `goglobal` authentication profile. The following sections will explain how to do this for each system supported by GO-

Global.

Solaris

On Solaris systems, PAM profiles are stored in the `/etc/pam.conf` file. You can create a basic `goglobal` authentication profile by adding lines such as these to the file:

```

goglobal    auth        required    /usr/lib/security/$ISA/pam_unix.so.1
goglobal    auth        required    /usr/lib/security/$ISA/pam_dial_auth
.so.1
goglobal    account     requisite    /usr/lib/security/$ISA/pam_roles.so.
1
goglobal    account     required    /usr/lib/security/$ISA/pam_projects.
so.1
goglobal    account     required    /usr/lib/security/$ISA/pam_unix.so.1
goglobal    password    required    /usr/lib/security/$ISA/pam_unix.so.1

```

Note that this configuration is an example only and may not directly apply to your system if your system does not use the default PAM configuration. Please consult your system documentation and/or system administrator if you have custom PAM requirements.

Linux

On Linux systems, each PAM configuration profile is placed in its own file in the `/etc/pam.d/` directory. You can create a basic `goglobal` authentication profile by creating a file in this directory called `goglobal`, containing the following:

```

auth        required    /lib/security/pam_securetty.so
auth        required    /lib/security/pam_stack.so service=system-
auth
auth        required    /lib/security/pam_nologin.so
account     required    /lib/security/pam_stack.so service=system-
auth
password    required    /lib/security/pam_stack.so service=system-
auth
session     required    /lib/security/pam_stack.so service=system-
auth
session     optional    /lib/security/pam_console.so

```

Note that this PAM configuration refers to a `system-auth` configuration, and essentially inherits most functionality from it.

Because of the large number of Linux systems and Linux configuration files, we recommend examining an existing profile, such as `login`, and begin constructing your `goglobal` profile from the existing one. Note that the `other` profile is the default for applications that do not have a specific profile, and that this profile typically denies all authentication attempts.

HP-UX

On HP-UX 11 systems, authentication profiles are stored in the `/etc/pam.conf` file. Create a basic `goglobal` authentication profile by adding lines such as these to the file:

```

goglobal    account     required    /usr/lib/security/libpam_unix.1
goglobal    auth        required    /usr/lib/security/libpam_unix.1
goglobal    password    required    /usr/lib/security/libpam_unix.1
goglobal    session     required    /usr/lib/security/libpam_unix.1

```

Disabling PAM

The GO-Global Server also supports basic authentication against the system password files (`/etc/passwd` and/or `/etc/shadow`). To disable PAM support and use the system password files, modify the `gold.conf` `auth` value to `auth=pass`. In this mode, gold will not use the PAM (or other modular authentication) API, but will instead authenticate using the `getpwnam` system calls to read the system password files directly.

Some authentication features, such as expiring passwords, will work in this mode, but unlike the modular authentication mechanisms, there is no way to change an expired password. For this reason, we do not recommend using this mode.

00020409

For more information, consult the following web site:

<http://www.kernel.org/pub/linux/libs/pam/modules.html>

Index

A

Adaptive Tuning, 20
aix, 12
Authenticating, 29
Authentication Methods, 12
Automatic proxy configuration, 34

B

Backing Store, 19
Borders, 43
Broadcast, 23
broadcast mode, 45
Browser client, 27
Button 4 / 5, 21

C

CA certificate, 58, 60
CA Key, 61
Cache Compression, 21
Cache Size, 21
CDE application, 16
Central License Server, 24
Certificate Authority, 58, 60
Certificate Signing Request, 61, 62
Certificates, 58
Client Back Buffer, 20
Client Display Settings, 21
closeapp, 44
closeconnection, 44
Closing a Connection, 32
Command-line options, 11, 43
Compression, 42, 44
Connecting, 28
Connection, 28
Current Connections, 30

D

Display, 21
Display Depth, 23
Display Name, 44
Display Options, 21
Display Size, 21

E

Enter Data, 7
Environment Variables, 11
Error logs, 47
Exceptions, 34
Extra Server Options, 21

F

File Transfer, 35
FLEXnet Publisher, 7, 24, 25, 49, 51
Font Path, 19

G

gdbi.log, 47
GO Window Manager, 16
GO-Configure, 15, 18, 29, 45, 49
GO-Global Application Configuration, 16
GO-Global Application Configuration program, 18
GO-Global Directory, 7
GO-Global License Manager Daemon, 10
GO-Global Login Daemon, 7, 9, 12, 48
GO-Global User, 7
goglobalux Script, 12
gold, 7, 9, 11, 12, 48
Gold log file, 49
gold.conf, 14
GraphOn Login Daemon, 11

H

Host ID, 9
Host Name, 9
hostid, 65

I

Installation log, 47
Installing the GO-Global Server, 6
Internet Explorer, 34

J

Java Swing Optimization, 20
joinsession, 44

K

Keytables Path, 19

L

LAN Settings, 34

Leader Log, 48

License Error Codes, 51

License File Method, 67

License Management, 24, 49

License manager log files, 12

License Server, 24

Licensing, 9

lock_path, 14

Log Files, 47

M

Macrovision, 24

Manual proxy configuration, 34

master, 65

Master Log, 47

Memory requirements, 5

Middle mouse emulation, 43

Minimize to system tray, 43

Mouse Compression, 20

Mouse wheel, 21, 43

Multidepth, 19, 46

Multiple Connections, 32

Multiple-window mode, 16

N

NFS Environment, 14

NFS Share, 14

Non-Default Ports, 11

Non-Privileged, 7

O

Obscurity Messages, 20

OpenSSL toolkit, 59, 60

Optimizations, 19

otherdb, 14

P

pam, 12

PAM, 68

Panning, 43

pass, 12

password, 44

pid_path, 14

Pluggable Authentication Modules, 68

Pointer warping, 43

port, 43

Processor requirements, 5

Product Code, 7, 9

proxy server, 33

Proxy Values, 33

Publishing applications, 14

R

Recent Connections, 28, 30

Redundant License Servers, 65

Registration, 9

root directory, 12

RSA technology, 61

Running Browser Clients, 33

Running Sessions, 30

S

Save Passwords, 42, 46

Screen Size, 43

Security, 46

Server, 6

Server Address, 29

Server Back Buffer, 20, 46

Server certificate, 59

Server key certificate, 59

Server Keys, 62

Server System Requirements, 5

Session, 30

Session frame, 43

Session Log, 48

Session logs, 48

Setting Proxy Values, 33

Single-window session, 43

Site Certificate Authority, 60

SSL, 46

SSL certificate, 65

SSL implementation, 59

Starting a Session, 29

Stopping a session, 31

Suspended Sessions, 30

Suspending a session, 31

syslog, 48

System requirements, 4

T

TCP/IP, 44, 46

Technical support, 57

three-server redundancy, 65

Time zone, 68

transport, 44

Trusted Server Certificate, 59

U

UNIX browser client, 26

Up / Down Arrow, 21

username, 43

W

Windows browser client, 26
Windows Clients, 26

X

X applications, 29
X desktop sessions, 29

X Display Manager Control Protocol, 45
X window manager, 16
xdm, 48
XDM, 45
XDM Host, 23
XDM Mode, 23
XDMCP, 45
XDMCP Host, 23