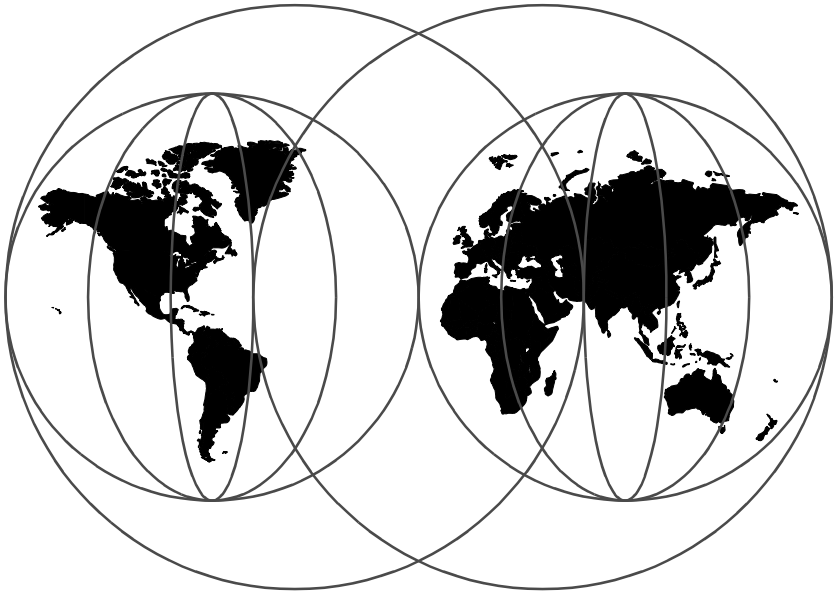
Sign On with IBM’s Global Sign-On!

Heinz Johner, Praveen K. Chandrashekhar, Patricia Savage, Klaus-Thomas Schleicher



International Technical Support Organization

http://www.redbooks.ibm.com

SG24-5122-00

SG24-5122-00

International Technical Support Organization

Sign On with IBM’s Global Sign-On!

November 1998

Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix E, “Special Notices” on page 291.

First Edition (November 1998)

This edition applies to IBM Global Sign-On for Multiplatforms, Version 2.0, 5697-GS2.

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. JN9B Building 045 Internal Zip 2834
11400 Burnet Road
Austin, Texas 78758-3493

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Figures

Tables

Preface

Despite the many obvious advantages of distributed applications and systems over centralized, heavyweight mainframe solutions, there are still some noteworthy considerations. User accounts and security management are two of the major concerns in such a heterogeneous environment. The IBM Global Sign-On for Multiplatforms, Version 2.0 product (hereinafter also called GSO) is a secure, distributed logon and password coordinator that helps administrators and users reduce this management burden.

This redbook will help you install, tailor and configure GSO. The redbook gives a broad understanding of the architecture, building blocks and the features provided by the product, and it discusses solutions to some typical scenarios that apply to most installations.

The first two chapters introduce GSO to readers who are not familiar with the product or who need to know the new features that come with Version 2.0. The chapters that follow explain the steps that are involved for exploitation of GSO, including planning, installation, customizing, and managing a production environment. Additional reference information is provided in the appendixes.

This book was written for technical consultants, designers and administrators involved in security planning and installations who need to know the concepts, technical architecture, and implementation of GSO.

The Team That Wrote This Redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization, Austin Center.

Heinz Johner is an Advisory Systems Engineer at the International Technical Support Organization, Austin Center. He writes extensively on the Distributed Computing Environment (DCE) and related areas. Before joining the ITSO, he worked in the Professional Services organization of IBM Switzerland and was responsible for DCE and Systems Management in medium and large customer projects.

Praveen K. Chandrashekhar is a Technical Executive for IBM Global Services in India. He has two years of experience in the Internet and intranet field. He has worked extensively on designing security solutions for the prospective Internet Service Providers in India. His areas of expertise include

IP network designs, implementation of security solutions and integrated solutions for corporate intranets.

Patricia Savage is an above-country pre-sales technical support specialist for EMEA. She has five years experience in this role and has supported the Global Sign-On product in the field for the last two years. She first joined the IT industry in 1976 and has worked at IBM for 13 years. Her areas of expertise in IBM include Transaction Server, the Distributed Computing Environment and GSO.

Dr. Klaus-Thomas Schleicher is an IBM Consultant I/T Architect in Germany. He has eight years of experience in the distributed computing environment field. He holds a doctorate of mathematics from the Technical University in Darmstadt, Germany. His areas of expertise include client/server infrastructure design and systems management.

Thanks to the following people from the GSO development team in Austin, Texas, for their invaluable contributions to this project:

Rich Caponigro
Mike D. Crane
George Dever
Jody Hasten
Tony Lai
Kazuko Maeda
Mark Molnar
Kevin O'Leary
Dave Schneider
Andrea Snow-Weaver
Don Williamson
George Wilson

Special thanks go to the editors for their help in finalizing the text and publishing the book:

Marcus Brewer
Tara Campbell
Elizabeth Barnes

Comments Welcome

Your comments are important to us!

We want our redbooks to be as helpful as possible. Please send us your comments about this or other redbooks in one of the following ways:

- Fax the evaluation form found in “ITSO Redbook Evaluation” on page 307 to the fax number shown on the form.
- Use the electronic evaluation form found on the Redbooks Web sites:

For Internet users	`http://www.redbooks.ibm.com`
For IBM Intranet users	`http://w3.itso.ibm.com`

- Send us a note at the following address:

  `redbook@us.ibm.com`

Chapter 1. Global Sign-On: The Global View

The advances in information technology in the past two decades have drastically improved business process automation and increased peoples' productivity. New applications that are used by several users at the same time with a shared set of data are as common as the telephone on everyone's desk.

While the advantages of new application concepts in conjunction with networked computers are obvious to end users, information technology specialists and corporate management, they bring an additional level of complexity along with them that, unfortunately, create some undesired overhead. Such overheads include:

- An increased complexity for management of such systems, requiring advanced management tools and highly skilled system administrators
- An increased complexity on the users' desktops due to multiple, less-than-perfectly integrated systems

The second of these overheads has a parasitic effect on user productivity. The "dream" of having just a single application with a consistent user interface for all the users' needs remains an architectural challenge for the coming years and decades. This would eliminate a number of headaches caused by today's multi-application solutions, such as data consistency, end-user education, systems management, and security.

Security in a distributed environment, which involves user authentication, data integrity and other security disciplines depending on the needs, is a major concern for corporate security officers. For end users, security systems most often are perceived as being additional burdens that create the necessity of maintaining and remembering multiple logon IDs and passwords and using them several times every day just to be able to perform their normal work. This is what might be called the "multiple-sign-on dilemma."

1.1 The Multiple-Sign-On Dilemma

Most users of desktop information technology equipment (most likely a Personal Computer) use that device for accessing a number of services, be it several local applications or more sophisticated applications that involve one or more remote system(s) to which the user's machine is connected through a network. Because security is a general concern, many applications require a user to authenticate before he or she is granted access to the services and data the application provides. Authentication is the process of ensuring that

the user attempting to access a service is actually the user he/she claims to be. This is usually done by requiring a user to enter a user ID and a (secret) password that is not known by anybody else. More modern systems may include in the authentication process a verification of a user's fingerprint or may require a user to insert a personal identification card (Smartcard).

The dilemma that arises is that a user has to maintain all these user IDs and passwords. It is fairly common that a single user has five or more such user accounts, all on different platforms with different rules for password lengths and patterns as well as change frequency. It is up to the user to either memorize them all or write them down, thus exposing a security risk.

Users not only have to memorize (or write down) their user account information, they have to change passwords regularly, as dictated by the rules given by either corporate standards or by the applications. With more passwords to be memorized, the chances increase that passwords are forgotten, which requires application (or platform) administrators to reset such passwords.

All in all, a measurable amount of time is lost every day because users have to log onto a number of systems and provide their passwords before they can do their work.

How Much Does Logon Cost an Organization?

In an organization whose annual salaries were an average of $50,000, the estimated logon costs were calculated to be in a range from somewhat less than $1,000,000 to more than $10,000,000, according to a position paper titled *Enterprise-Wide Security: Authentication and Single Sign-On* published by the Network Application Consortium (NAC,

IBM Global Sign-On for Multiplatforms, Version 2.0 provides a secure, easy-to-use solution that grants users access to the computing resources they are authorized to use—with just one logon. Designed for large enterprises consisting of multiple systems and applications within heterogeneous, distributed computing environments, GSO eliminates the need for end-users to manage multiple logon IDs and passwords.

GSO, from a high-level perspective, works as follows (more details can be found in Chapter 2, "Global Sign-On: The Macro View" on page 13):

1. A user logs on to an operating system using a user ID and a password.
2. If GSO is configured for integrated logon, it uses that user ID and password and logs the user on to GSO. This is called primary

authentication. It uses a highly secure network authentication mechanism that involves a GSO server.

3. GSO then presents a list of available targets the user can logon to. Depending on the individual configuration, GSO can automatically sign the user onto his/her targets after authenticating to GSO or let the user select which targets he/she wants to sign-on to.
4. The user may choose to change the password on any such target through the GSO user interface. GSO carries out that function out and stores the new password for subsequent use.
5. After the user has finished working with a target platform, GSO logs the user off (either on specific request or when that user chooses to log off from GSO).

The key design point here is that GSO securely stores and uses user IDs and passwords for each target platform and user. This approach allows for a maximum of flexibility because target platforms do not need to be adapted or changed.

1.2 High-Level Design Goals of GSO 2.0

The desire for a single sign-on solution emerged when users began logging onto more than just one operating system, remote system(s), or application(s) and discovered the lost time and aggrivation associated with remembering multiple IDs and passwords. IBM Global Sign-On for Multiplatforms, Version 2.0 addresses the issues involved in multiple user signons with the following conceptual solutions:

- Most target platforms use (and require) their own proprietary authentication mechanism:

  GSO 2.0 does not replace proprietary authentication mechanisms. Thus, applications are not required to be changed in order to benefit from GSO.

- Most target platforms have their own user ID and password rules:

  GSO does not impose any new rules; it supports whatever rules are demanded by the target platforms.

- Handling multiple user IDs and passwords per user is an administrative nightmare:

  GSO securely stores user account information (such as user ID, password, and some additional information) in a database per target platform and per user. This information is retrieved from that GSO

database and used to log a user on to a target platform. By doing this, the user is relieved from handling and memorizing user IDs and passwords.

- Managing user accounts among multiple platforms creates additional administrative load (and cost):

  GSO integrates user management for GSO and the target platform in one single tool: Tivoli User Administration. Through a single user interface, a user account can be managed on a large number of target platforms.

- The availability of a centralized authentication system decreases due to a dependency on a single server:

  GSO supports multiple, replicated servers that can dynamically spread the load. A single server can go off-line without disrupting the authentication service. This provides an excellent basis for an uninterrupted, highly available service.

- A single sign-on solution needs to be managed:

  GSO 2.0 integrates into the Tivoli Management Environment (TME). Common processes, such as server and client installation, configuration, administration tasks, system monitoring, and user administration are all integrated into the Tivoli framework. This also eliminates or minimizes the need for administrator education.

- How is security affected by a single sign-on solution?

  GSO uses a highly secure authentication and communication platform underneath that uses Kerberos authentication for primary authentication and DES (Data Encryption Standard) encryption. Sensitive user information is stored and transmitted over the network only after being encrypted.

- User authentication should be standardized among platforms.

  At the time most platforms (operating systems, network operating systems, applications, and so on) were developed, there was no common standard available and thus most vendors came up with their own proprietary authentication mechanisms. Only recently was the X/Open Single Sign-On (XSSO) standard introduced. Through its modular concept, GSO supports the XSSO standard.

- A considerable amount of help desk work load is caused by problems with user passwords.

  GSO drastically reduces the rate of forgotten passwords since it stores the users' passwords safely and securely.

1.3 The User's View of Global Sign-On

In 1.1, "The Multiple-Sign-On Dilemma" on page 1, a high-level flow of actions on how GSO works was described. This section briefly outlines what GSO does from a user's perspective.

First of all, GSO will most likely operate in a configurable mode that is called *integrated login*. When so configured, GSO gets the user ID and password from the operating system (Windows 95 or Windows NT only) at the time the user logs on to that operating system. Because of this, no additional logon to GSO is required in this mode. On OS/2 Warp, or if integrated login is not configured or when the user signs off from GSO and then restarts GSO without logging off the operating system, a simple dialog window pops up for user sign-on to GSO, as shown in Figure 1.

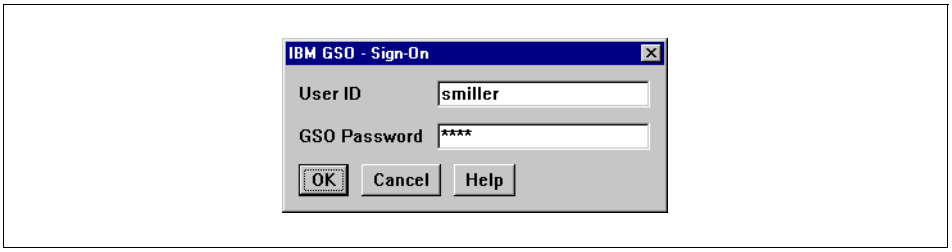


Figure 1. GSO Login Window

Note: It should be mentioned at this point that there are additional methods for authenticating a GSO user. These are through the use of fingerprint readers and/or Smartcards. These is covered later in this book.

The user ID and password that need to be entered are the user's GSO account information. Every user is registered with GSO, but in most environments, this user ID and password are chosen to be the same as the ones used for logon to the operating system (which is a requirement when integrated login is used).

Provided that a valid GSO user ID and password was entered, the GSO Launcher will be presented to the user, as shown in Figure 2.

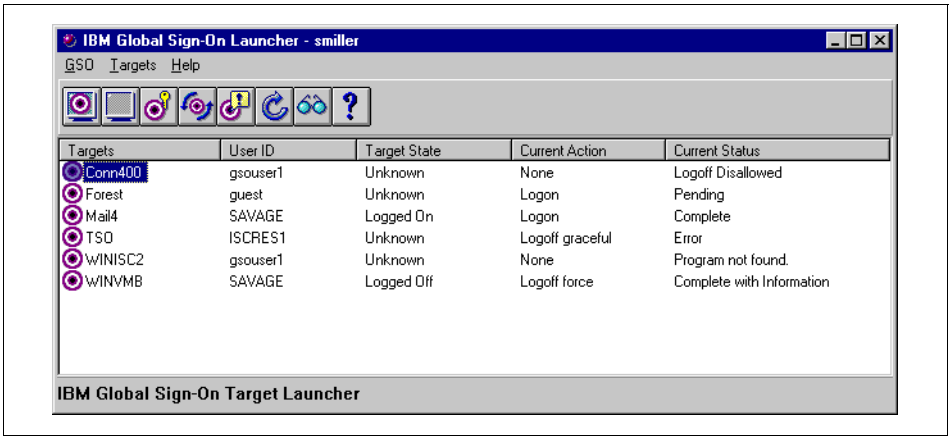


Figure 2. GSO Launcher Window

The GSO Launcher is the primary interface that GSO provides for the user. It presents the user a list of target systems that he or she can access, the logon ID used on each target, and some status or error information for each target. The target definitions as shown on the Launcher window are actually stored on a GSO server, not locally on the user's system. Because of this, a user may log on at a different machine and still find the same list of targets on the Launcher window.

Through the Launcher window, GSO also offers a set of functions to the user. These functions are accessible through pull-down menus, and the most important ones are also accessible directly by clicking on their shortcut icon. These functions include:

- Change the GSO password. The GSO password is used when logging on to GSO, either explicitly through the login window (see Figure 1) or implicitly when using integrated login.
- Logging on to and logging off from selected targets. The user chooses a target from the list and either selects the logon or the logoff function. The Current Status column in the Launcher window is updated accordingly.
- Change the password for a selected target. After selecting a target from the list, the user's password for logging onto that target can be changed. Note that not all targets support password change operations (Figure 3).

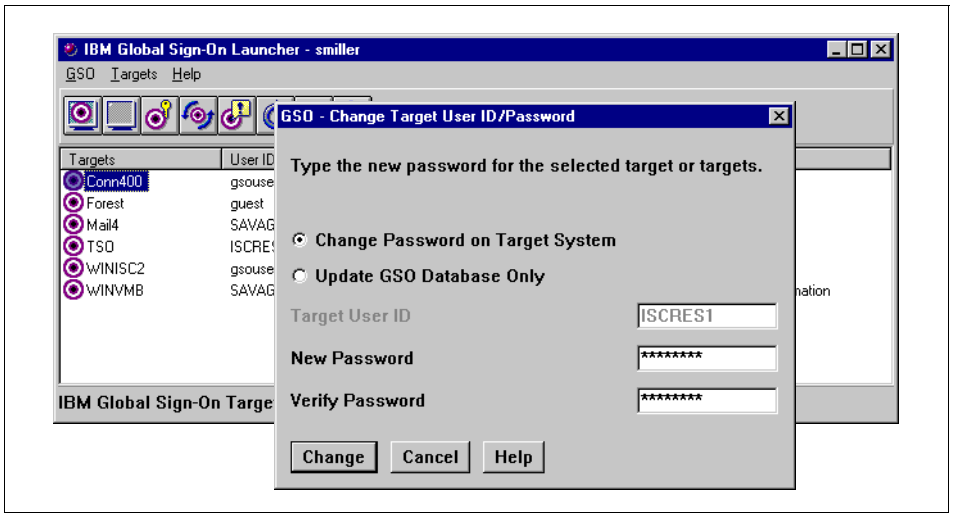


Figure 3. Changing a Target Password

- Change the logon preferences for each target. Such preferences specify, for example, whether or not the target should be logged on to automatically when logging on to GSO or how the logoff process should be performed (forced, graceful, or not allowed at all).
- Browse log files. GSO maintains log files that can be useful for problem determination.
- Refresh the displayed status of the targets shown in the Launcher window.

In addition to the Launcher window described above, the user can start the GSO Administration window. This is not normally used for daily tasks, but may be helpful or even necessary for client system configuration tasks (as described in Chapter 5, “Installing GSO Clients” on page 103 and Chapter 6, “Defining Targets” on page 147).

A sample Administration window is shown in Figure 4. It shows a list of programs that are defined and configured for that user. A program can be thought of as a method that is used to log on to a target. Besides some general functions for changing the GSO password and logoff, the functions that are available through the Administration window are all geared to manage these programs. In most cases, users (and even administrators) do not need to work with the Administration window after these programs have been properly set up on a user’s workstation.

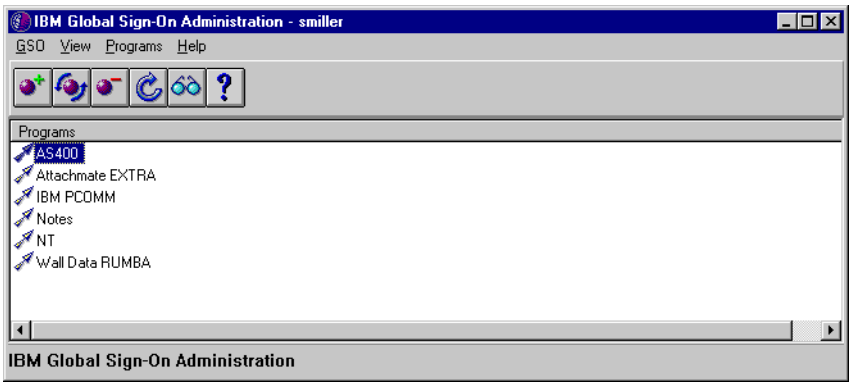


Figure 4. GSO Administration Window

It should be mentioned for completeness that the user also has access to a set of GSO online documentation, such as the Launcher and Administration Help books, a Command Reference, a Programmer's Guide, and the Readme file, after GSO is installed on the workstation. The documentation can be accessed through the **Start** menu on Windows or through the **IBM Global Sign-On Client** icon on the OS/2 Warp desktop.

1.4 The Administrator's View of Global Sign-On

Although this section is not meant to be an administrator's guide to GSO, it might be worthwhile for the introduction to and understanding of GSO to look at some typical tools and tasks an administrator uses to manage a GSO environment.

As mentioned earlier, GSO administration is integrated into the Tivoli Management Environment (TME). This includes server and client installation and configuration, user administration, systems monitoring, and execution of several common administration tasks. It is assumed that GSO is installed in a running Tivoli environment and that the administrators are familiar with the concepts and use of Tivoli. This environment should exploit the Tivoli Framework (always required), Tivoli Software Distribution, Tivoli User Administration, Tivoli Distributed Monitoring, and optionally the Tivoli Enterprise Console. If this is not the case, you should consider getting some training on these topics first before installing GSO.

Because of this tight integration with Tivoli, GSO administration does not require much additional training once the Tivoli environment is well

understood. The integration with Tivoli is two-fold: the GSO Plus module adds a large set of new functions to the Tivoli Framework, Tivoli Software Distribution, and to Tivoli Distributed Monitoring, and the TME 10 GSO User Administration extends the Tivoli User Administration to be able to manage GSO user accounts.

1.4.1 Installation and Configuration Aids

Through the use of Tivoli Software Distribution in conjunction with the GSO Plus module, GSO servers and clients can be easily installed and configured. File packages for the various server and client roles, as well as for the supported platforms, can be created in preparation of a distribution task. Creating a GSO file package is made easy by the existence of dialog windows and task scripts specific to that task. Once file packages have been created, they can be distributed to the designated systems, either one at a time (maybe appropriate for servers) or to multiple systems at the same time.

After the file packages have been installed, GSO supports the configuration of servers and clients through ready-made tasks. These tasks require a minimum of information on a dialog window and then carry out the GSO configuration of the specified system(s), be it GSO servers or clients.

Installation and configuration of GSO servers and clients is described in Chapter 4, “Installing GSO Servers” on page 61, and in Chapter 5, “Installing GSO Clients” on page 103, respectively.

1.4.2 Administration Task Library

GSO comes with a rich set of ready-made tasks that can be executed against single or multiple systems. The screen shot in Figure 5 gives you an idea of what tasks are available. Other tasks in the configuration task library (not shown here) are provided for configuration of servers, clients and event adapters.

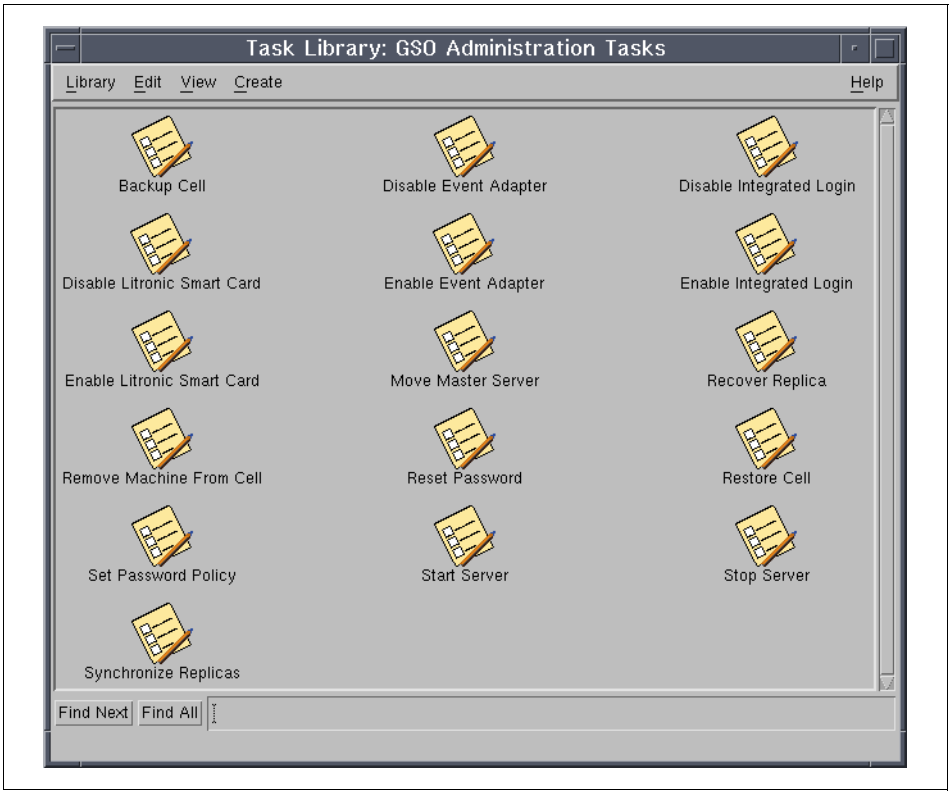


Figure 5. GSO Task Library

The tasks allow an administrator to perform specific actions on remote systems without requiring the administrator to remotely log in to those systems. They also simplify administration because many of these tasks actually execute programs that do more that just execute a single command.

More on these tasks can be found in the installation and configuration chapters that follow and, specifically, in 7.1, “GSO Management Tasks” on page 205.

1.4.3 User Administration

Tivoli User Administration supports a number of different platforms, for example UNIX, Novell NetWare or mainframe systems. Through the installation of TME 10 GSO User Administration, GSO becomes an additional supported target that a user administrator can manage with the user interface or through commands he or she is already familiar with. Moreover, since a user account for each user most likely exists on multiple platforms (thus the

need for GSO), this user account can be managed from one single point; Tivoli provides management integration, and GSO adds security integration of multiple user accounts for each individual user.

GSO user administration is explained in more detail in Chapter 8, “Managing User Accounts” on page 225.

1.4.4 Systems Monitoring

GSO Plus adds a number of monitors to the Tivoli Distributed Monitoring application. These include:

- Surveillance of GSO-related processes on GSO server machines
- Monitoring utilization of critical file system space
- Monitoring specific file sizes for critical files, such as log files
- Monitoring paging space

These monitors are preconfigured for the specific needs of GSO, for example the file systems that GSO utilizes. Events are generated and displayed on the administrator’s console when any parameter falls out of specified limits. Additional monitors can be added using standard mechanisms provided by Tivoli.

Through the use of the Tivoli Enterprise Console (TEC), event handling can be further improved by filtering and spreading events to multiple administrators, or by automatically running tasks in response to an event.

GSO monitoring is further described in Chapter 7, “Managing GSO” on page 205.

Chapter 2. Global Sign-On: The Macro View

The first chapter introduced you to the IBM Global Sign-On for Multiplatforms, Version 2.0 (hereafter referred to as GSO) product from a global view, what it does and how it helps you to improve productivity while maintaining a high level of security in a heterogeneous system and application environment. This chapter goes into more details and explains the GSO components consists of from a more technical point of view. Reading this chapter is strongly recommended in order to understand the terminology that is used throughout the reminder of the book and to learn how the components interact with each other.

The first section that follows gives you an overall picture of GSO, and the subsequent sections explain the pieces in more detail.

2.1 Overview

GSO relieves the user from having to type different user IDs and passwords for all his or her target systems, which includes operating systems, groupware solutions, databases, or almost any other kind of application. It is important to understand that most such targets do not provide a programming interface or any other means such that GSO could do the authentication in lieu of the authentication services contained within those targets.

As you might have noticed, the term *target* was just introduced. As explained above, in GSO terminology, a target is whatever a user wants to log on to, be it a database application or any other application that requires a user ID and a password. To be more correct, targets are those applications and services that GSO supports—that is, the places where GSO can do the logon for a user. GSO itself, of course, is not considered a target since the user has to do the log-on.

The ideal world would exist if GSO could act as the only highly secure, trusted authentication mechanism that any target would rely on. Unfortunately, it is not possible to design and develop such a single authentication (or single sign-on) solution because most products that incorporate an authentication service do this in a different way and do not provide the published interface that would be required. Vendors would have to modify their products to adhere to a common standard, such as the X/Open Single Sign-On (XSSO) standard, to make such an ideal world a reality.

As explained in Chapter 1, “Global Sign-On: The Global View” on page 1, GSO takes another approach—in fact, the only feasible approach given the

fact that vendor products do not support trusted external authentication. For authentication, these products typically require (at least) a user ID and a password for each user. GSO takes care of this by storing user IDs and passwords in a secure manner and providing them to the targets when the user wants to sign on. This relieves the user from having to remember and enter these IDs and password every day for every target.

Let's take a look at Figure 6 below. It shows the basic components that we need to understand before we move on. The user interacts with his or her workstation and some applications (targets) that may be running on this workstation or on another computer, such as a departmental server or a mainframe. This is all common business, were it not for the GSO server.

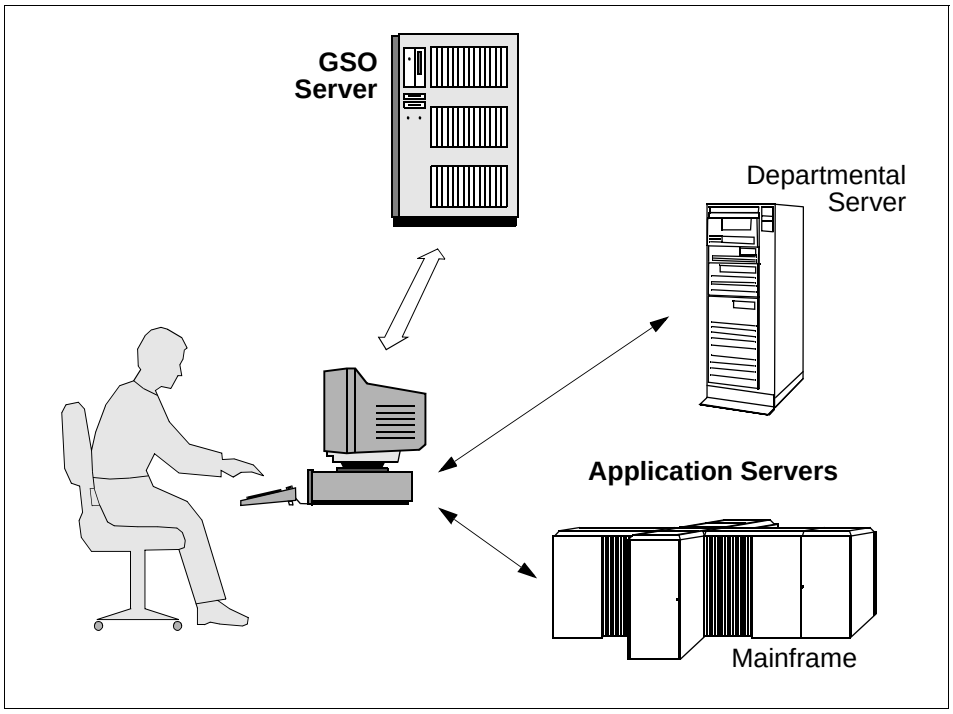


Figure 6. GSO - The Basic Picture

The user most likely has to log onto his/her workstation before any work can be done (Windows 95 does not necessarily require a user to log on). What changes now is that the user signs on to GSO, rather than to any of the applications or other servers. GSO does the authentication based on a user ID and password, maybe supported by a Smartcard or a fingerprint reader. The GSO server is involved in this authentication process in order to verify

the user's password and to get his/her credentials. GSO, not the user him/herself, will then log the user on to all the other applications and servers (targets) that this user is supposed to work with. GSO uses the methods provided by the targets to log the user on. In most cases, GSO simulates a user log-on by providing a user ID and a password to the target as if the user had entered them. The big difference, obviously, is that the user does not need to remember all these user IDs and passwords; GSO takes care of them.

Figure 6 shows the basic layout of a *GSO cell*. A GSO cell consists of at least a GSO server and at least one user workstation, also called a *GSO client*, that form an administrative unit. There can be more than one GSO server and as many as thousands of clients—still considered a single GSO cell.

The following sections explain in more detail what the GSO clients' and servers' roles are within a GSO cell.

2.2 GSO Clients

As shown above, GSO is a client/server application, which means that, in addition to the GSO server, there is a piece of code running in the user's workstation that interacts with the GSO server. In fact, this GSO client code has many different functions to carry out, which are explained throughout this section. In a first step, some high-level building blocks (shown in Figure 7) are briefly introduced.

Later in this section, the function blocks shown in Figure 7 are further examined and taken apart, but for the time being, we concentrate on this higher level.

2.2.1 The Raw Picture

Figure 7 depicts the major pieces in a GSO client. It should be noted at this point that there is also a specialized type of GSO client, called a GSO *database client*, which runs some database client code in addition to what is shown here. GSO database clients are discussed in 2.2.3, "Database Clients" on page 19.

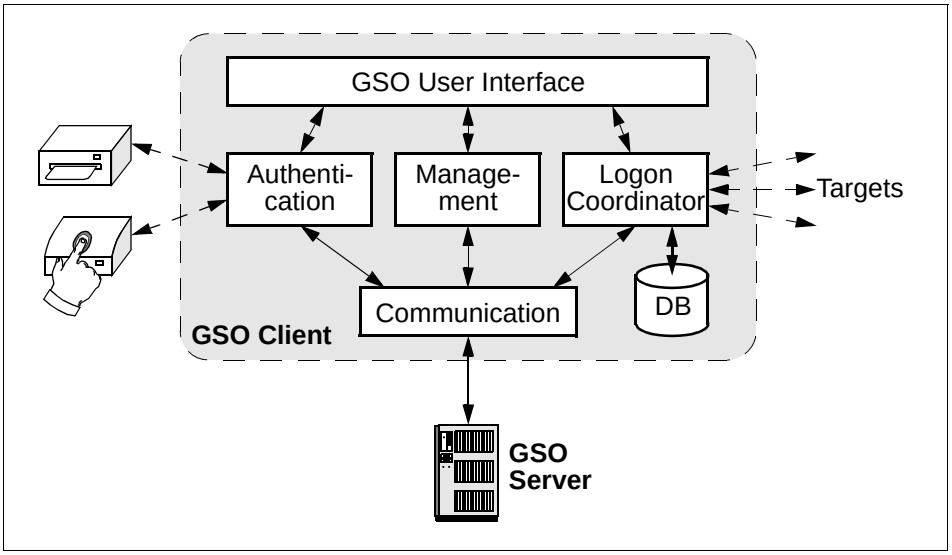


Figure 7. GSO Client Function Blocks

The GSO *user interface* is responsible for all interactions with the user (and/or administrator). Most interactions are done through a graphical user interface (GUI), although it should be mentioned that there is also a command line interface (CLI) for most functions. The latter is convenient, or even essential, when GSO functions are to be carried out by other programs that normally cannot interact with a graphical user interface.

The *authentication service* is a very important part of GSO. It is responsible for reliable user authentication, and it does this in one of three ways:

- By prompting the user for a user ID and a password
- By using Smartcard as an additional authentication means in conjunction with a Smartcard personal identification number (PIN)
- By using a biometric device (fingerprint reader) as an additional authentication means

In any case, the authentication service interacts with the GSO server to verify the user's ID and password and to get additional user information. The GSO server is a required component for successful user authentication.

The *management services* allow the user to change passwords, change his or her target definitions, and to tailor other functions according to his/her needs. The administration services interact with the GSO server to store and/or alter the user's definitions stored in the GSO server.

After successful authentication, the *logon coordinator* finally performs the logons to the various targets. It gets the user's target information and, more importantly, the user's IDs and passwords for the respective targets, from the GSO server. The logon coordinator is supported by a local database that stores the client's target information and the methods used to log the user onto the targets.

Communication between the GSO clients and the GSO servers uses highly secure encryption (DES, Data Encryption Standard) and other means that will be discussed later to prevent the GSO installation from security attacks on the network.

Now that the basics of a GSO client have been explored, the pieces can be taken apart further to look a bit closer at them in the next section.

2.2.2 The Details

Figure 8 unveils the functional blocks that can be found in a GSO client. They can best be understood when we follow the flow of actions as a user logs on to GSO and to the targets.

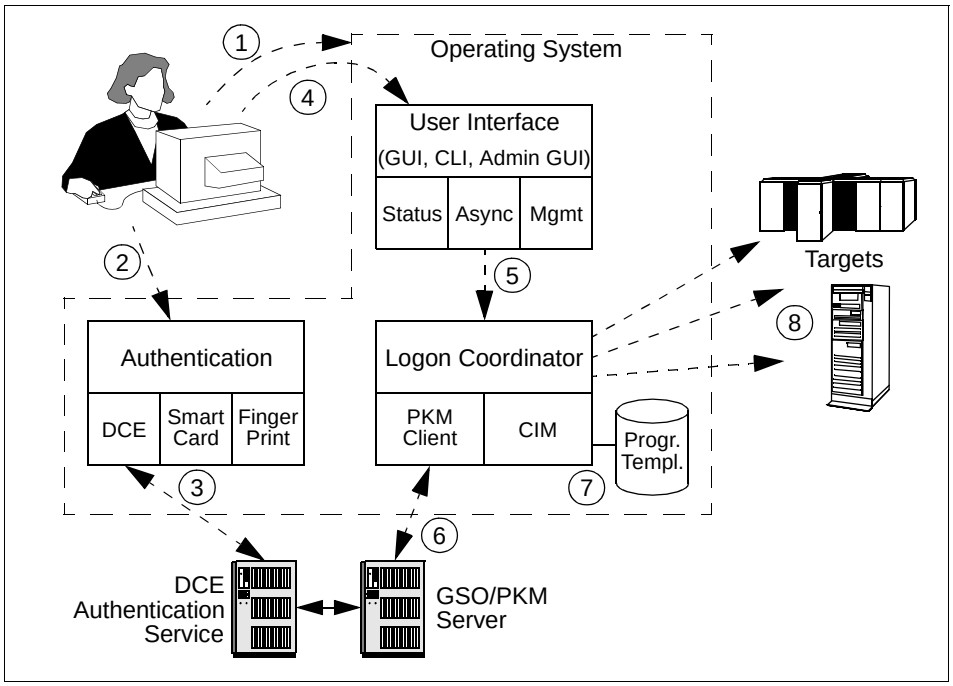


Figure 8. Detailed View of a GSO Client

First of all, it might be surprising that there seems to be two GSO servers in Figure 8. This is true as far as actual server services is concerned, but in practice they almost certainly run in one single machine. GSO servers are discussed in 2.3, “GSO Servers” on page 20.

Back to Figure 8, the steps when a user logs on are:

1. The user logs on the operating system, if required (Windows 95 does not necessarily require this).
2. The user logs on to GSO by providing his or her user ID and a password. Depending on the individual setup, it might be required that the user uses a Smartcard or places a finger on a fingerprint reader for successful authentication.
3. The GSO authentication services uses a highly secure authentication mechanism based on the Distributed Computing Environment (DCE) to verify the user’s user ID and password.

   DCE was chosen for GSO because it is a proven network authentication system based on Kerberos that ensures that there are no passwords transmitted over the network.
4. After successful authentication, the user is presented a list of available targets to log on to. Alternatively, depending on the specific configuration, GSO might log the user on to several targets automatically.

   It should be mentioned at this point that the GSO GUI also offers other services, as depicted in Figure 8. For example, the user can display the actual logon status or view and modify certain preferences.
5. The user can select a target from the list and request GSO to log him or her on.
6. For each individual target, the logon coordinator (LC) requests and receives the pertinent data, such as target type, user ID and password, from the Personal Key Manager (PKM). All data transfers over the network are encrypted.
7. The Configuration Information Manager (CIM) then retrieves additional information from a local database. Such information is specific to each target type and includes, for example, the method to be used to log the user onto that target.
8. Using the correct methods determined in the last step, the logon coordinator calls the related program that performs the authentication with that particular target. This finally logs the user onto that target.

The steps for logging a user off from a target, or to change a password, are basically the same as steps 5 through 8 above.

The GSO data model, as mentioned in the steps above, relies on two distinct sets of data: the PKM data objects in the central PKM server store user-specific information, while the CIM data objects on each GSO client store configuration-specific information.

The GSO Configuration Information Manager needs a bit closer attention because it may involve some local configuration and administration on the user's client machine. The CIM manages the configuration information of the target logon mechanisms configured for the machine. For each target, a *program template file* (PTF) is required that specifies the access methods for that particular target. For example, a PTF contains information about where a target application is installed on a GSO client and which program (method) is to be called to actually perform the logon, logoff and password change operations. Different targets require different methods. Targets can be classified into target types that have similar properties, such as MVS systems or database applications. Such target types are defined in schema files. Several PTFs and schema files are shipped with GSO for the targets that are supported. It is also possible that new PTFs and/or schema files can be added in order to support new targets (see 6.7, "Adding New Targets to the GSO Framework" on page 199).

The PTFs are clear text files that could be edited manually using a standard text editor. In order to make them usable for the CIM, they have to be converted to a machine-readable format, and they are then called programs. Programs (in GSO terminology) are machine-readable forms of PTFs and schema files which the CIM internally uses to perform its operations.

A last term, the Logon Script Files (LSF), deserve some mentioning. GSO supports logon capabilities to 3270/5250 type applications. This is done using a process called screen scraping for 3270/5250 sessions. An LSF can be thought of as a small logon program that contains the sequence of actions, both user input and expected host response, for a 3270/5250 logon process.

2.2.3 Database Clients

A GSO database client is actually a normal GSO client that has an additional GSO database client module installed. The type of module depends of the database that is being used: ODBC (for IBM DB2, Informix, Oracle, and Microsoft SQL Server), OCI (for Oracle), or CT-LIB (for Sybase). The additional database module that resides on a database client interacts with the database server (see 2.3.2, "Database Servers" on page 22) to perform

GSO user authentication with the database (RDBMS) running on the GSO database server.

2.2.4 Summary

Table 1 summarizes the software components that run in a GSO client. It is for your reference only and does not mean that these components need to be installed and maintained separately. For more details on how to install and configure GSO clients, please see Chapter 5, “Installing GSO Clients” on page 103.

Table 1. GSO Client Software Components

Client Type	Component	Remarks
GSO Client	GSO Client	GSO client code shown in Figure 7
	DASCOM NetSEAT DCE Client	Windows 95 and NT only; contains the minimal run-time environment to support the GSO client (not a full DCE client)
	IBM DCE Client	For OS/2 Warp only
GSO Database Client Support (Same as above, plus one of the database clients shown to the right)	ODBC Database Client	ODBC Client for Windows 95 and NT
	OCI Database Client	Oracle OCI Client for Windows 95 and NT
	CT-LIB Database Client	Sybase CT-LIB Client for Windows 95 and NT

Table 1 shows the DCE client (either DASCOM or IBM DCE client) as a separate component. In case of OS/2, this has to be installed and configured separately, but in case of Windows (95 and NT), the DCE client is integrated into the GSO client and does not need any separate attention. If you need to do some configuration beyond a standard GSO setup (for example as discussed in Appendix A, “Extended Configuration Methods” on page 237), you should understand that the DCE client can be configured separately.

2.3 GSO Servers

So far, the GSO server has been introduced as a black box service, providing whatever the clients request. Figure 8 on page 17 revealed some more details and separated the DCE authentication services from the PKM service. This section takes a closer look at the GSO server.

2.3.1 The Details

The GSO server has two primary functional goals:

- User authentication–Performs the primary user authentication. Once successfully authenticated, the GSO client can retrieve all other user information for that user.
- User configuration data management–Also called Personal Key Manager (PKM), this service stores the user IDs and passwords, along with some other information, for the targets in a secure way, inaccessible for other users.

Another service incorporated into the GSO server is the passticket generator. Passtickets, also known as one-time passwords, can be used to log a user on to a host that uses RACF as the authentication system, provided RACF is customized to accept passtickets.

The Rationale Behind Passtickets

What is the advantage of one-time passwords? They were introduced to overcome a security exposure that traditional 3270 type terminals or PC emulators had because they transmitted users' passwords in the clear over the network. Passtickets are generated with a certain algorithm based on a shared secret key, the time of day and other information. They can only be used once and are only valid for a relatively short period of time. Passtickets are worthless to an intruder who has hooked up to the network with a trace tool.

The GSO server has some secondary goals, too. Because it is a mandatory server, needed for the GSO clients to work properly, provisions need to be provided for availability, scalability, manageability, and reliability. This is achieved by the following means:

- Availability–GSO servers can be replicated; that means, there can be more than one operational GSO server. Should any of these servers fail, clients can connect to another to get the information they need. In such a replicated environment, only one GSO server can be a master server, while all others are replica servers. Updates, such as password changes, can only be done through the master server. However, any replica server can take over the role of a master server if it becomes necessary, for example when the (former) master server fails for an extended period of time. All such operations, such as addition or removal of a replica server, can be done without service interruption.

- Scalability–Replication not only eliminates single points of failure (and thus increases availability) but also provides a perfect means for scalability. Client access load can be distributed to several servers, resulting in a (theoretical) unlimited growth potential. If a server needs to be replaced, for example by a more powerful one, replication allows this to be done without any service interruption.
- Manageability–A GSO server does not require a complicated setup procedure, nor does it require daily maintenance. In normal operation, a GSO server hardly ever requires any maintenance tasks to be performed on it. Installation can be done through ready-prepared steps using the Tivoli Management Environment (TME), as explained in Chapter 4, “Installing GSO Servers” on page 61.
- Reliability–The GSO server is implemented on a proven, secure and reliable framework, the Distributed Computing Environment (DCE).

The DCE middleware framework, on which GSO builds, provides most of the availability and scalability advantages of GSO. It also ensures highly secure user authentication and storage of sensitive data in the security registry, which is protected by DCE access control lists (ACLs).

2.3.2 Database Servers

GSO database servers are machines that are part of the GSO cell that run some special piece of code. This code assures GSO database clients (see 2.2.3, “Database Clients” on page 19) have authenticated access to an underlaying relational database management system (RDBMS) which can either be IBM DB2, Informix, Oracle, Sybase, or MS SQL Server. Although not required, it is recommended that GSO database servers run on the same physical machine as the RDBMS itself.

GSO database servers and GSO servers (see last section) have little in common in terms of functionality and should not be confused.

2.3.3 Summary

To summarize, Table 2 lists all the components that make up a GSO server or a GSO database server, respectively. Bear in mind that these components

need not be installed, configured and maintained individually; they are listed for reference information only.

Table 2. GSO Server Software Components

Server Type	Component	Remarks
GSO Server	DCE Security Server	May be configured as Master or Replica server
	DCE Directory Server	May be configured as initial or additional directory server
	DCE Distributed Time Service (DTS)	A DTS (local) server is needed
	DCE Client	Includes the dce.pthreads fileset
	SMIT panels	DCE and GSO
	DASCOM Directory Service Broker	
	DCE Privacy Package	Contains the DES encryption library
	IBM Global Sign-On Server	
	DCE Tivoli Event Console Adapter	
GSO Database Server	DCE Client	Includes the dce.pthreads fileset
	DCE Privacy Package	Contains the DES encryption library
	SMIT panels	DCE and GSO
	DCE Tivoli Event Console Adapter	
	IBM GSO ODBC Database Broker Server, or IBM GSO OCI Database Broker Server, or IBM GSO CT-Lib Database Broker Server	Depending on the RDBMS product

For information on how to install and set up GSO servers, including GSO database servers, please see Chapter 4, “Installing GSO Servers” on page 61.

2.4 Supported Platforms

Mentioned below are the platforms on which the GSO servers and clients can be installed. The basic Tivoli software levels that are supported are also mentioned.

GSO 2.0 installation comprises of the following modules:

- TME 10 GSO User Administration 3.1
- Tivoli/Plus GSO module
- GSO server(s) depending on the number of GSO servers that are planned to be installed
- GSO client(s) depending on the number of nodes to be installed
- If there is a database server which is to be placed under GSO, then one or more GSO database server(s)
- If there is a GSO database server installed, then one or more GSO database client(s)

Note

You should always read the README files and other documentation that come with the product for latest information about supported platforms and software levels. The information given here is for your information only and can be outdated by the time you receive the actual product package.

2.4.1 GSO Master and Replica Server

You can install the GSO Server, GSO User Administration, GSO Plus module, and the GSO Database Server on any of the following platforms:

- IBM AIX, Version 4.2.X
- Sun Solaris, Version 2.5.1
- Microsoft Windows NT, Version 4.0 with ServicePak 3

Tivoli 3.2 is the basic prerequisite for the GSO installation. Ensure that a Tivoli environment exists at the following levels:

- TME 10 Framework 3.2 (including patch 3.2-TMF-0007)
- TME 10 User Administration 3.1.3
- TME 10 Software Distribution 3.1

There are various functions provided by TME 10 as a suite. In order to take complete advantage of them in the GSO environment, ensure that they are available at the following level:

- TME 10 Distributed Monitoring:
  - TME 10 Distributed Monitoring 3.5.1 (including patch 3.5-SEN-0005)
  - TME 10 Distributed Monitoring Universal Motors
  - TME 10 Distributed Monitoring UNIX Monitors
- Enterprise Console:
  - TME 10 Enterprise Console 3.1
  - TME 10 Enterprise Console 3.1 Server

2.4.2 GSO Database Server

The GSO Database server can be installed only on those platforms supported by the GSO server. However, the databases that are supported are:

- DB2 2.1.1.2
- Oracle 7.X
- Sybase 10.X
- Informix 7.X
- MS SQL Server 6.X

2.4.3 GSO Clients

Mentioned below are the platforms that are supported as GSO clients and as GSO database clients.

GSO 2.0 supports the following operating system platforms as clients:

- Microsoft Windows 95
- Microsoft Windows NT 4.0 with ServicePak 3
- IBM OS/2 Warp 3.0 (with FixPak 21 or higher) or Warp 4.0

IBM OS/2 Warp client for GSO Version 1.5 is included (shipped) with GSO 2.0. There is no GSO 2.0 client software for OS/2 Warp.

2.4.4 GSO Target Systems

The GSO software supports the following targets:

- Novell NetWare Server 3.12 and above
- IBM OS/2 LAN Server

- IBM OS/2 Warp 3 Server
- IBM Client Access for AS/400 for Windows 95/NT
- Microsoft Windows NT 4.0
- Lotus Notes 4.5 and above
- 3270 host systems, including those protected by RACF and those that recognize passtickets when used with the following emulators:
  - IBM Personal Communications AS400/3270 for OS/2 Warp, Microsoft Windows NT, and Windows 95
  - Attachmate EXTRA! for Windows 95 and Windows NT
  - Wall Data RUMBA for Windows 95 and Windows NT
- 5250 host systems when used with the following emulators:
  - IBM Personal Communications AS400/3270 Emulator for Microsoft Windows NT and Windows 95
  - Attachmate EXTRA! for Windows 95 and Windows NT
  - Client Access/400 for Windows 95 and Windows NT
- SnareWorks (IntelliSoft Corp.)

2.5 Hardware Requirements

Hardware requirements for GSO differ for each operating system platform. Mentioned below are a classification of the same.

Note

Please read the README files and the documentation that came with the GSO product for latest information on hardware requirements. The information provided here is for your convenience and for planning purposes only and may be outdated by the time you receive the product.

2.5.1 GSO Master and Replica Server

Table 3 lists the hardware minimums for GSO servers for a basic, small installation. For large installations, please also read the sizing section in 3.3, “GSO Server Machine Recommendations” on page 42.

Table 3. GSO Server Hardware Requirements

Platform	Disk Space (Permanent)	Disk Space (Temporary)	Memory
AIX	35 MB	130 MB	64 MB
Solaris	100 MB	110 MB	64 MB
Windows NT	60 MB	140 MB	48 MB

Bear in mind that the values provided for memory in Table 3 need to be adjusted if additional applications or tools run at the same time.

2.5.2 GSO Database Server

Table 4 shows the minimum requirements for GSO database servers.

Table 4. GSO Database Server Hardware Requirements

Platform	Disk Space (Permanence)	Disk Space (Temporary)	Memory
AIX	45 MB	50 MB	64 MB
Solaris	120 MB	15 MB	64 MB
Windows NT	70 MB	40 MB	64 MB

Memory and disk requirements largely depend on the size of the database (and other applications) that run on the same server. The requirement mentioned above are for GSO components only and do not take into account any database or any other application.

2.5.3 GSO Clients

The requirements for GSO clients and GSO database clients are listed in Table 5. There will most likely be other applications running on clients, and

thus, the values for memory need to be adjusted accordingly to represent the total amount of memory required for a client machine.

Table 5. GSO Clients Hardware Requirements

Platform	Disk Space (Permanent)	Memory
Windows 95	20 MB	16 MB
Windows NT	20 MB	24 MB

OS/2 Warp client machine requirements depend largely on the various options that you have when installing DCE for OS/2. As a rule of thumb, GSO (excluding DCE) on OS/2 Warp requires about the same resources as on Windows.

2.6 Target Support

Following is a list and short description of the targets that GSO 2.0 supports. For each target, the methods used for authentication and password changes are briefly described.

3270 and 5250 Emulation

For 3270- and 5250-type terminal applications, GSO uses a method called screen scraping by calling the Enhanced High Level Language API (EHLLAPI) of these terminal emulation programs. This method is commonly used by processes that automatically interact with host applications through such a terminal emulator on a workstation. The method basically reads the host screen and provides user input into the correct fields on the screen, expecting the next screen, and so on. Additional Logon Script Files (LSFs) must be provided that carry out the logical sequence of such a logon, password change, and logoff (see also 2.2.2, “The Details” on page 17).

For further information, see 6.4.9, “3270 and 5250 Emulation” on page 182.

Client Access/400

Client Access/400 is another terminal access product for IBM AS/400. From a GSO point of view, Client Access/400 works the same way as 3270 and 5250 e7mulation as described in the last section.

For further information, see 6.4.4, “Client Access/400” on page 176.

Novell NetWare

Logon to a Novell NetWare network is supported for Windows as well as OS/2 Warp clients. Support is provided for logon, change password, and logoff. The NetWare APIs are utilized for these functions.

For further information, see 6.4.2, “Novell NetWare” on page 173.

Microsoft NT Server

Logon to a Microsoft NT server is supported for Windows 95 and Windows NT; OS/2 Warp clients can access an NT server though their LAN Server client support. It uses the net function API for logon, password change and logoff. Logging on to a NT server allows a user to utilize shared devices, such as disks or printers.

For further information, see 6.4.3, “Windows NT 4.0” on page 175.

Lotus Notes

Lotus Notes client supports a sophisticated authentication feature that GSO takes advantage of. Through a minor configuration change, Notes does not require the user to enter his or her user ID and password in a pop-up window, but expects a program, such as GSO, to provide this information on behalf of the user. This way, GSO integrates ideally into Lotus Notes using this authentication feature.

For further information, see 6.4.1, “Lotus Notes” on page 171.

IBM LAN Server

The IBM LAN Server has much in common with the Windows NT Server logon. Though it is only supported on OS/2 Warp, the Windows system can log on to a LAN Server domain using the NT Server logon capability described above. The requester’s API functions are utilized for logon, change password and logoff.

For further information, see 6.4.5, “LAN Server Logon to a Domain” on page 178, 6.4.6, “LAN Server Local Logon” on page 180, and 6.4.7, “LAN Server Manage Passwords in a Domain or on a Server” on page 181.

Databases

GSO user authentication to databases is done by means of database server and client modules. This technology, developed by Open Horizon, provides transparent network access to relational database management systems

(RDBMSs) from applications that exploit those databases. It consists of a server and a client part that provides standard APIs to applications while performing user authentication through GSO. Such support is available from GSO for ODBC (Open Database Connectivity), OCI (Oracle Call Interface), and CT-LIB (Client Library). Databases that are supported using these interfaces include IBM DB2, Informix, Oracle, Sybase, and MS SQL Server.

Databases as GSO targets are not covered in more detail in this redbook.

SnareWorks

SnareWorks from IntelliSoft Corp. is a suite of products that provide legacy applications a high degree of access security. This is done by intercepting the TCP/IP traffic between specified clients and servers and adding the required security to it. This way, the applications do not need to be changed, nor do they need to be aware of any security at all. SnareWorks adds these features on the communication layer. At the time a SnareWorks client attempts to access a resource that is protected by SnareWorks, a window pops up asking the user for a valid user ID and password. SnareWorks then checks with its database whether or not this is a valid access and will then either grant or deny that access. Any other TCP/IP traffic on the server or client systems is not affected by SnareWorks.

GSO can do automatic user authentication and change of passwords for SnareWorks as a target.

For further information, see 6.5, “Implementing SnareWorks” on page 186.

PeopleSoft

Support for the PeopleSoft application suite is provided through the database authentication methods available for Oracle.

PeopleSoft as a GSO target is not covered in more detail in this redbook.

2.7 Integration with Tivoli

The Tivoli Management Environment (TME) suite of products is IBM’s strategic management platform for comprehensive systems, security, application, network, and inventory management. It consists at least of the Tivoli Management Framework (TMF) and a set of either Tivoli or third-party management applications that run on that framework.

GSO 2.0 is tightly integrated with Tivoli and basically requires that a Tivoli environment is already in place before GSO can be installed, configured and employed. GSO takes advantage of the Tivoli framework in various areas:

- Software distribution, installation and configuration
- User administration
- Monitoring and notification

These topics are discussed in the sections that follow.

2.7.1 Software Distribution and Configuration

Tivoli Software Distribution is a Tivoli management application that is used by GSO for distribution and configuration of GSO servers, GSO database servers, GSO clients, and GSO database clients. The CD-ROMs that come with the GSO 2.0 product package include complete file packages for these GSO machine roles that can easily be set up and distributed using the Tivoli Software Distribution. Following a distribution job, the basic setup is automatically performed such that there is little or nothing left to be done on those systems.

Installing GSO servers and clients using Tivoli Software Distribution is covered in Chapter 4, “Installing GSO Servers” on page 61, and in Chapter 5, “Installing GSO Clients” on page 103, respectively.

2.7.2 User Administration

Tivoli User Administration can be considered a superset of all user administration tools that exist on the various systems it supports. For example, Tivoli User Administration supports the user properties required for UNIX users, NIS (yellow pages) users, Windows NT users, and so on. Figure 9 depicts a sample collection of such supported systems at the bottom.

Through the extensions of the TME 10 GSO User Administration module, Tivoli User Administration becomes the primary user administration tool for GSO. In other words, GSO becomes a supported endpoint, or managed resource, for Tivoli User Administration. In Figure 9, these extensions are shown as shaded areas. In particular, TME 10 GSO User Administration extends the Tivoli User Administration in three areas:

- Graphical User Interface: New entry fields on the dialogs allow for entering and changing of GSO-related data.
- User Attribute Database: The additional, GSO-related attributes are stored in the Tivoli database.

- Managed Resource: GSO becomes a new managed resource for user profiles, just like the many others that might be there already.

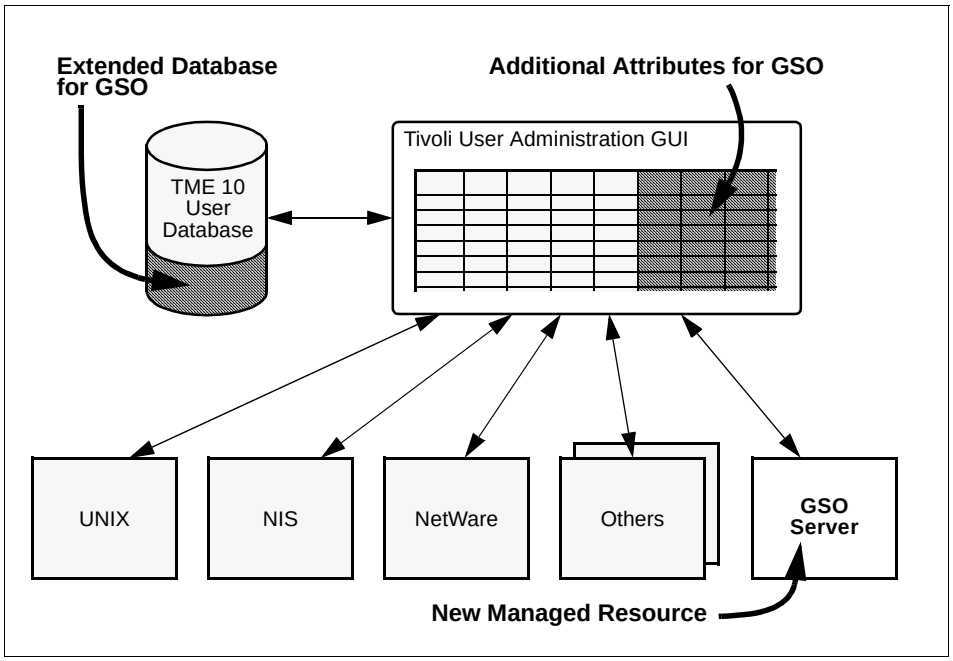


Figure 9. GSO Extends Tivoli User Administration

In order to have these extensions to Tivoli User Administration available, the TME 10 GSO User Administration module needs to be installed onto the Tivoli Management Framework. See also 8.1, “Integration in Tivoli User Administration” on page 225, for more details.

2.7.3 Monitoring and Notification

Tivoli Distributed Monitoring (DM) provides a rich set of framework functions and sample monitors to monitor systems and applications and alert administrators of certain conditions. Together with the Tivoli Enterprise Console (TEC), an administrator or operator can have a complete view of the status of server machines, server processes, conditions, and other critical resources to prevent these services from failing.

See Chapter 7, “Managing GSO” on page 205, for details on what you can monitor, how you can be notified of certain conditions, and how to set up monitoring and notification in a GSO/Tivoli environment.

Chapter 3. Planning for Global Sign-On

In order to deploy the IBM Global Sign-On for Multiplatforms, Version 2.0 to a production environment, one has to set up a deployment plan that identifies responsibilities and tasks for everyone participating in the project, for example, end-users and system administrators. The deployment plan should minimize the impact of the change-over on systems administration, on the production system, and on the business routine. Creating a good deployment plan helps to identify the risks and issues related to the rollout of GSO in a production environment. Depending on the target environment, the deployment plan may cover:

- The education and training schedule for each of the participating audience. For example, system administrators have to be trained to use the TME 10 GSO User Administration component, and end-users have to be trained to use the GSO user interface.
- Setup and migration periods—the schedule for setting up GSO servers, GSO clients, configuration of GSO resources, migration (or integration) of the current user administration processes, and migration of the login process for end-users.
- Design and implementation of the GSO cell—the number and layout of GSO servers and clients.
- A list of all items or system characteristics, such as Network, Tivoli, service-level agreements and their validation, and processes that must be in place before a change-over can occur. This list serves as a base for defining checkpoints.
- Defining checkpoints and their validation during the deployment process.
- Defining a support plan during and after the change-over period. This plan may contain backup and recovery procedures, problem determination and diagnostic procedures.
- Detailed instructions for the deployment team on how to configure the different components, such as the Tivoli environment, GSO servers and clients.

Some of the tasks described above, for example education and the definition of a support plan, are very dependent on the target environment. Therefore, in the following section, we discuss the design and implementation of the GSO cell. Other parts of the deployment plan can be derived from the chapters that follow or from the product documentation available for IBM Global Sign-On for Multiplatforms, Version 2.0 and the Tivoli products.

3.1 The Environment

This section describes different scenarios that IBM Global Sign-On for Multiplatforms, Version 2.0 supports and the resulting GSO cell layout. Depending on the topology of the target environment, it may be appropriate to have several GSO cells. The description focuses on the implementation of GSO; it is assumed that the corresponding Tivoli environment is already in place (some additional Tivoli integration issues, however, are discussed in 3.7, “Tivoli Integration” on page 48). Although technically possible (and in small installations likely), neither the TMR server nor the target systems have to be part of the GSO cell. In the following discussions, the TMR servers and the target systems are considered and depicted as being outside the GSO cells.

Four scenarios are selected and described as examples for typical implementations:

- A minimal configuration supporting a small to medium number of users, where planned and unplanned outages of the GSO services are acceptable. Planned outages may occur for backup and archive purposes and for hardware or software maintenance; unplanned outages may be introduced by hardware, software or network malfunctioning. This scenario is described in “Minimal Configuration” on page 35.
- An implementation that handles a large number of users and provides a reliable login services to end-users. Other services, for example GSO user administration and password change, can be restored within view minutes using Tivoli tasks. These tasks may be triggered by administrators, or may run as automated tasks. This example scenario is described in “Scalable Configuration with Highly Available Login Service” on page 37.
- An implementation that handles a large number of users and provides high availability for all services using IBM’s High Availability Cluster Multi-Processing/6000 (HACMP/6000), which automates the takeover procedures and minimizes service down-times. See “High Availability Configuration” on page 39.
- Implementation within an existing DCE environment. This also covers the coexistence of a GSO and a DCE client on the same machine and may be useful for environments where DCE and/or DFS is already deployed. See “Implementation in an Existing DCE Cell” on page 41.

Within each scenario, the placement of server and client components is described. Please refer to Chapter 2, “Global Sign-On: The Macro View” on page 13, for a more detailed introduction and description of these components.

Note

The components listed in the following sample scenarios for the various machines are listed for your reference and to help you understand GSO. Users and administrators usually do not have to be concerned about all these components because most of them are installed and configured automatically and do not require maintenance or other administrative work.

3.1.1 Minimal Configuration

The minimal configuration of a GSO cell consists of one GSO server that is managed by a Tivoli Management Region (TMR) server and a number of GSO clients, as shown in Figure 10. If required, the GSO server can also act as a GSO client. Optionally, database services can be included in this scenario to take advantage of GSO's user authentication capabilities.

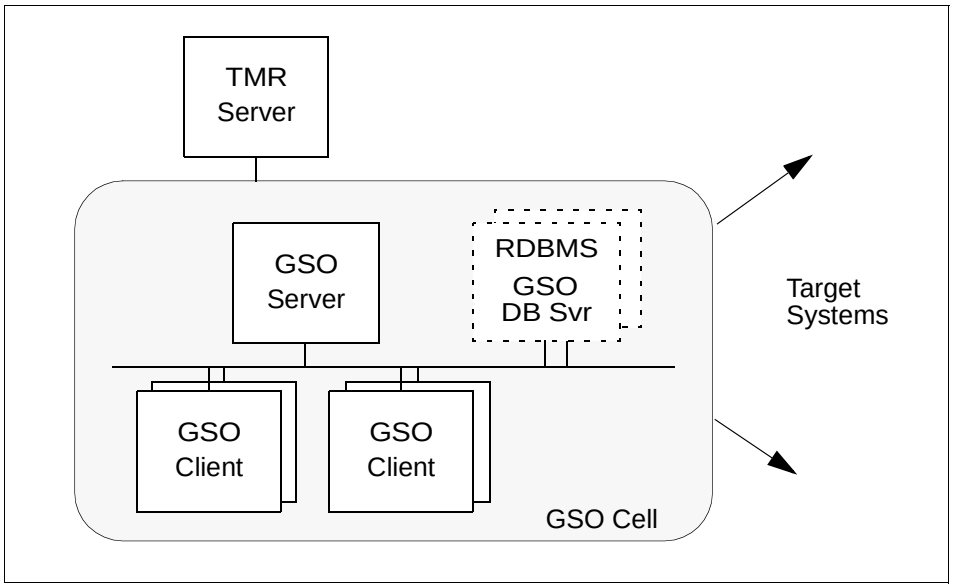


Figure 10. Basic GSO Cell Configuration

The GSO server hosts the following components and services:

- Tivoli Managed Node, including:
  - Tivoli Framework
  - Tivoli User Administration
  - Tivoli Software Distribution (optional)
  - Tivoli Distributed Monitoring (optional)

    - Tivoli/Plus GSO Module
    - TME 10 GSO User Administration Module
- GSO Server Package, including:
    - DCE Security Server (Master)
    - DCE Cell Directory Service
    - DCE Distributed Time Server
    - DASCOM Directory Service Broker (DSB)
    - GSO Server
    - DCE Client

If GSO is used to authenticate users to database targets, it is recommended that the GSO database server runs on the same system as the target RDBMS. The GSO database server hosts the following components and services:

- DCE Client
- GSO Database Server, including:
    - GSO OCI DB Server, if Oracle Call Level Interface (OCI) is used
    - GSO CT-LIB Server, if Sybase is used
    - GSO ODBC Server, if an ODBC server (DB2, Informix, Oracle ODBC, Microsoft SQL) is used

The GSO clients host the following components:

- Tivoli Managed Node (not required, but recommended) or Tivoli PC Managed Node (not required, but recommended). This may include Tivoli Software Distribution and Tivoli Distributed Monitoring Modules.
- GSO Client Package, including:
    - GSO Client, including the DASCOM NetSEAT DCE client
    - GSO Client user interfaces, for example GSO login window

A minimum configuration, as described above, leverages all the basic function of GSO and may include a number of different targets. However, since there are no provisions for increased availability, such as replication, this kind of installation depends on single components, such as the GSO server. If the GSO server fails, users cannot log on to GSO or targets, and no administration tasks can be performed. To overcome this, replication of server services can be incorporated as described in the next example scenario.

An installation like this may be typical for test and education purposes, but due to its limitations, it should not be implemented where availability and/or

scalability is a concern, unless there are other means to overcome temporary outages.

3.1.2 Scalable Configuration with Highly Available Login Service

A scalable configuration of a GSO cell typically consists of one GSO master server (there can only be one master server), one or more GSO replica servers, and a number of GSO clients, as depicted in Figure 11. The GSO replica servers provide additional availability for the login service.

GSO clients are able to authenticate as long as they have network connectivity to at least one GSO server (master or replica). If the master is not available for an extended period of time, one can elect a GSO replica server to become a new master server using the methods described in 7.1, “GSO Management Tasks” on page 205, or using standard DCE administration procedures, thus preserving the capability to change passwords and to manage user accounts.

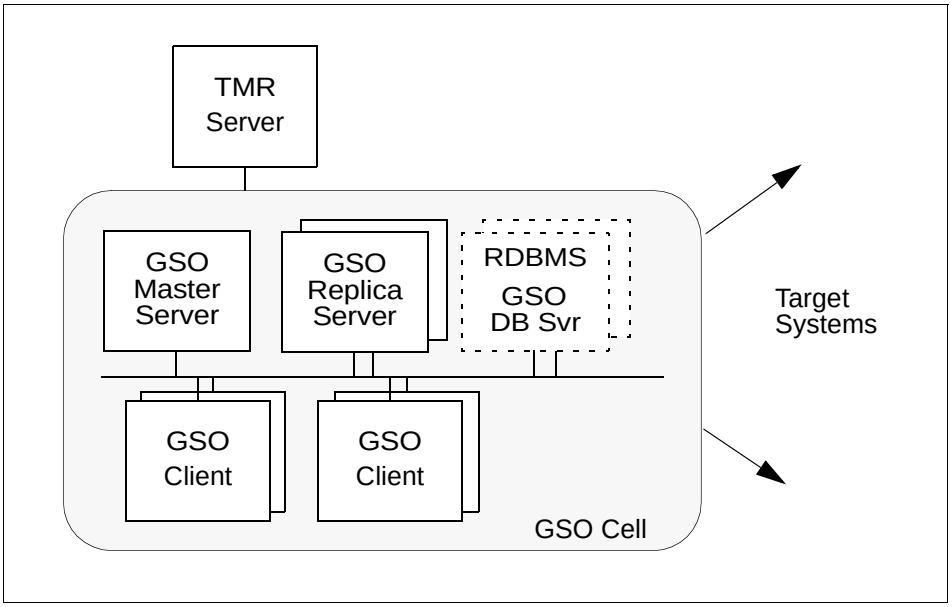


Figure 11. Scalable GSO Cell Configuration

The addition of GSO replica servers also enhances the performance, and therefore the scalability, of the GSO cell. In order to adapt to a changing workload, GSO replica servers may be added or removed without any service interruptions. The standard installation process for a GSO replica server configures a replica of each of the following: a DCE security server (secd), a

DCE directory server (cdsd), a Directory Service Broker (DSB), and a GSO server (gsod). However, should a particular installation require this, it is possible to only replicate, for example, a security server or a GSO server. In A.1, “Configuration of GSO Servers on an Existing DCE Cell” on page 237, we explain how GSO integrates with DCE if you need additional information.

The GSO servers and (optionally, but recommended) the clients are part of a Tivoli Management Region (TMR) managed by the TMR server.

The GSO master server in this configuration runs the following components and services:

- Tivoli Managed Node, including:
  - Tivoli Framework
  - Tivoli User Administration
  - Tivoli Software Distribution (optional)
  - Tivoli Distributed Monitoring (optional)
  - Tivoli/Plus GSO Module
  - TME 10 GSO User Administration
- GSO Server Package, including:
  - DCE Security Server (master)
  - DCE Cell Directory Service (primary)
  - DCE Distributed Time Server
  - DASCOM Directory Service Broker (DSB)
  - GSO Server

The GSO replica server runs the following components and services:

- Tivoli Managed Node, including:
  - Tivoli Framework
  - Tivoli Software Distribution (optional)
  - Tivoli Distributed Monitoring (optional)
  - Tivoli/Plus GSO module
- GSO Server Package, including:
  - DCE Security Server (Replica)
  - DCE Cell Directory Service (Secondary)
  - DCE Distributed Time Server
  - DASCOM Directory Service Broker
  - GSO Server

The GSO client and the GSO database server host the same software as listed in 3.1.1, “Minimal Configuration” on page 35.

In addition to the configuration explained in the first example (3.1.1, “Minimal Configuration” on page 35), this sample configuration replicates (duplicates) the DCE and GSO server services. In such a configuration, a single server may be temporarily unavailable for maintenance tasks or due to a system outage, but users continue to be able to log on to GSO and to GSO targets. Users may experience short delays caused by the change-over processes when using these services. Replication provides increased availability for read operations, but not for write operations, should the master server become unavailable because all write operations can only be performed by the master server. Logging on to GSO and to targets are read operations, where the users’ properties need to be queried only. Password changes and user administration, on the other hand, are typically write operations, where the users’ properties are being changed. Therefore, adding users or updating current information, including password changes, cannot be performed in case of a failure of the master server in this configuration.

There are ways to further increase the availability for write operations. An easy way is to set up (and practice) the procedures to elect a replica server to become a new master server. This takes only a few minutes (but it may take considerably longer to realize and identify a problem with the master server). Another way to improve availability with the capability of automatic fail-over can be provided with a scenario described in the next section.

3.1.3 High Availability Configuration

The scalable configuration of a GSO cell, as described in the previous section, can be extended by also providing a highly available user management service that would allow user administration even in the case of a master server outage. This can be achieved by using a high availability solution for the GSO master server. If, for example, an IBM AIX server is used for the GSO master server, High Availability Cluster Multi-Processing/6000 (HACMP/6000) for AIX can be used to provide this level of service (Figure 12). The configuration of the master server is the same as the implementation of a DCE high availability solution. Further information on this can be found in the *IBM DCE for AIX, Version 2.2: High Availability Cluster Multi-Processing Guide for DCE and DFS* (see Appendix F, “Related Publications” on page 293 for details). In addition to the procedures described there, there are two additional configuration considerations for GSO:

- The /var/gso file system (or directory) has to be on a shared disk.
- To automate the required fail-over and reintegration actions, the Directory Service Broker daemon (DSB) has to be started and stopped from the corresponding HACMP/6000 scripts. The following line should be added to the start script after the line where DCE is being started:

```
/usr/bin/gsocfg -start dsb
```

Add the following line to the stop script before the line where DCE is being stopped:

```
/usr/bin/gsouncfg -stop dsb
```

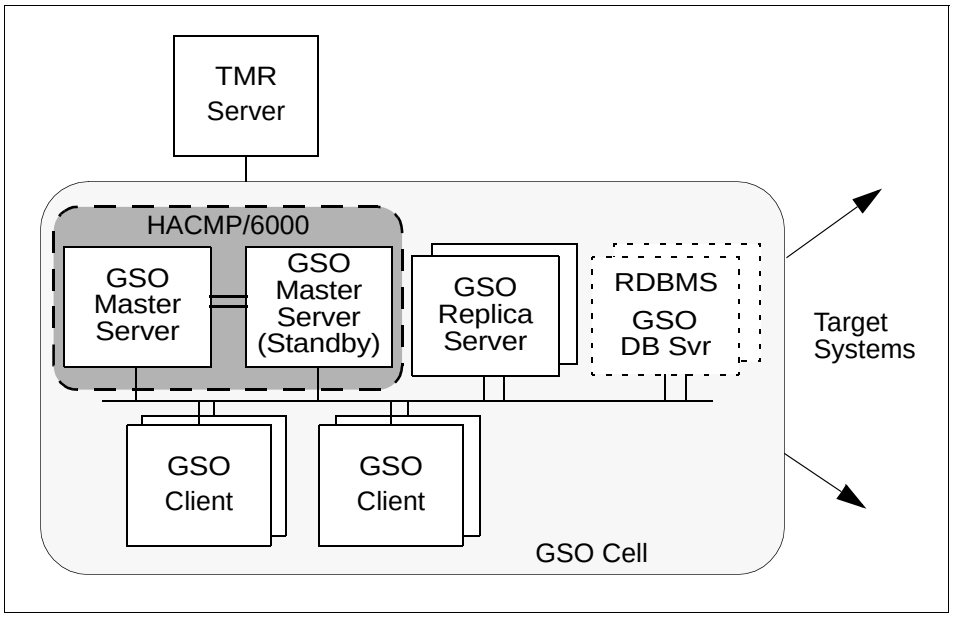


Figure 12. Using HACMP/6000 to Increase Master Server Availability

This scenario adds availability by adding redundant hardware and system functionality that is beyond the services normally provided by GSO. HACMP/6000 is designed specifically to add this extra level of availability to almost any kind of service, such as GSO.

Since HACMP/6000 does not provide an increase of performance (unless more powerful server machines are being used), the addition of replica servers may still be considered for scalability.

What is HACMP/6000?

HACMP/6000 is a clustering software from IBM that is designed to meet the high levels of availability required in business-critical applications. Among several supported operating modes, the most common mode is to have a pair of machines of which only one is active during normal operation, while the other runs as a hot stand-by system. Once HACMP/6000 detects a failure of the active machine, it can automatically connect the second system to the network and switch the application(s) over. Both systems share some disks that store application data. Depending on the service provided on these servers, such a fail-over can be absolutely transparent to clients.

3.1.4 Implementation in an Existing DCE Cell

IBM Global Sign-On for Multiplatforms, Version 2.0 uses a DCE infrastructure underneath to securely store and transmit sensitive authentication data. GSO makes this DCE cell transparent to the administrator and to the user because there are normally no operations necessary that directly relate to DCE, even during the installation of the product.

There might be, however, situations where a customer wishes to use an already existing DCE cell and install GSO such that it uses this cell, rather than creating its own cell. A standard installation of a GSO server using the Tivoli software distribution mechanisms automatically installs and configures DCE server services on the GSO server machine (or refuses installation if there is already a DCE server installed and configured); thus, modifications to the installation process need to be done.

Such modifications require a thorough understanding of GSO, DCE and the specific customer environment, and they might affect the product support agreement with IBM. Customers are therefore encouraged to request technical support through IBM Global Services in such cases.

For a more technical discussion on how GSO and DCE interoperate, as well as the installation process, please refer to A.1, “Configuration of GSO Servers on an Existing DCE Cell” on page 237, for details.

3.2 How Many GSO Cells?

One of the important design questions to answer is: How many GSO cells does one need within the target environment? Though the question is simple, there is no general answer to it. The optimal design will depend on the

topology of the target environment, performance and availability requirements, security policies, and organizational aspects. However, here are some considerations and guidelines that help to work out a cell design:

- Because of the administration overhead, it is generally a good idea to minimize the number of GSO cells. This is especially true if travelling users have to be supported and if one wants to avoid adding these users to more than one GSO cell.
- Policy regions within the Tivoli Management Environment allow management regions that are more granular than the scope of a GSO cell. Therefore, it is not necessary that administration domains are mapped one-to-one to GSO cells.
- More than one GSO cell can be managed by a Tivoli Management Region (TMR) server. The GSO User Administration endpoint and each GSO server has to belong to at most one TMR server.
- One GSO cell can support up to approximately 50,000 GSO users (this is not a hard limit, but rather an upper limit based on tests performed) as long as the network topology provides enough bandwidth, which is usually the case for a high-speed campus networks.
- If the environment is spread over different locations and the network connections between these locations are either unreliable or of low bandwidth, one may consider setting up multiple GSO cells. However, configuring one GSO cell and using GSO replicas in the locations or at major hubs can be sufficient.

When designing your GSO cell(s), please move on to the next sections, which provide some guidelines on sizing and hardware requirements for large GSO cells.

3.3 GSO Server Machine Recommendations

This section gives you some information that helps for sizing server machines. It is assumed that the GSO server hosts the DCE Security, DCE Directory, DCE Time, DASCOM Directory Service Broker, and the GSO services, which corresponds to a standard GSO installation.

The sizing of the server machines depends on:

- The total number of registered users. This is mainly because the DCE Security server keeps the DCE registry in memory, and therefore, a greater number of registered users will demand more memory on the DCE Security server.

- The number of concurrent logins.
- The number of targets per login.

The DCE registries on the GSO master and on all replicas store the same information that uses the same amount of memory. The client configuration determines the precedence in which GSO servers are contacted. Therefore, the client configuration provides a basic means for load balancing between the servers. If client requests are equally distributed to the available servers, the servers should experience approximately equal loads and therefore have similar capacities.

Tests have been conducted, and recommendations for sizing have been compiled in IBM's GSO test lab. IBM employees can access the results of these tests in the related white paper *Capacity Planning Guidelines* (by Andrea Snow-Weaver and Kazuko Maeda) on the IBM intranet at `w3.software.ibm.com/sales/networkingsw/globalsignon`. The assumptions for these sizings are that all users log in equally distributed within one hour and that average users have five targets defined.

Processor types are listed as examples for machines that can handle the workload. The GSO workload is typically CPU bound as long as there are no paging activities. The DCE and GSO server processes keep most data, for example the DCE registry, in memory. Therefore, sufficient memory should be installed to prevent the systems from paging activities. Table 6 provides an overview of memory and disk space requirements on GSO servers.

Table 6. Memory Requirements to Support a Given Number of Users

Number of Users	Number of Servers	Memory Size	Data Disk Space	Remarks
up to 4,000	2	128 MB (96 MB for NT)	100 MB	Configuration with one master and one replica server sized as in Table 7
4,000 to 10,000	1 server per 2,000 users	108 MB + 5 KB/user	60 MB + 10 KB/user	Machine sizing as in Table 7
10,000 to 52,000	1 server per 3,000 to 5,000 users	108 MB + 5 KB/user	60 MB + 10 KB/user	Machine sizing as in Table 8
over 52,000 users	Should be split into several GSO cells with less than 52,000 users each			

Small Environments

For small environments, a configuration with one GSO cell and at least two physical machines, a server and a replica is recommended. Each machine should be sized according to Table 7 in order to support up to 4,000 users. The environment can grow up to 10,000 users if memory and disk space is added according to Table 6. If the number of users is less than 2,000 and the availability requirements tolerate the absence of the login services while a server is down, a single server may be sufficient.

Table 7. Server Sizing for 4,000 Users

Operating System	Processor (Example)	Specint 95	Memory	Disk Space in MB		
				Product	Data	Temp
AIX 4.2	Power PC 604e 166 MHz	5-6	128 MB	35	100	130
Solaris 2.5.1	Ultra 1 Model 140	5-6	128 MB	100	100	110
NT 4.0	Pentium 200 MHz	5-6	96 MB	60	100	140

Note: For disk space location, see 3.5, “File System Layout and Space Requirements” on page 46.

Medium Environments

For medium environments, a configuration with one GSO cell and at least two physical machines, a server and a replica is recommended. Each machine should be sized according to Table 8 in order to support up to 10,000 users. The environment can grow up to 52,000 users if replicas and memory are added according to Table 6.

Table 8. Server Sizing for 10,000 Users

Operating System	Processor (Example)	Specint 95	Memory	Disk Space in MB		
				Product	Data	Temp
AIX 4.2	Power PC 604e 233 MHz or Power PC 604e 166 MHz SMP	12+ 7	160 MB	35	140	130
Solaris 2.5.1	Ultra 1 Model 140	9	256 MB	100	140	110
NT 4.0	Pentium 200 MHz	12+	160 MB	60	140	140

Note: For disk space location, see 3.5, “File System Layout and Space Requirements” on page 46.

Large Environments

It is recommended that large environments, having more than 52,000 users, should be split up into multiple GSO cells. Sizing can then be done according to the descriptions and tables given above, which are provided for small to medium environments.

A Word About Memory and Disk Requirements

The figures for memory and disk requirements listed in the tables above have been observed in a specific lab environment. Most installations have other applications, such as administration or monitoring tools, running at the same time. The values provided above should therefore be considered as starting points for your planning. The actual installed physical memory should be large enough such that GSO servers do not have paging activities. Since disk space is relatively inexpensive nowadays, it is always a good idea to have sufficient free disk space available beyond what your calculations indicate.

3.4 Client Machine Recommendations

The sizing of client machines depends more on the applications that are running on the client than on the GSO components. Table 9 shows space and memory recommendations for a GSO client. Applications may have higher requirements.

Table 9. Client Memory and Space Requirements

Operating System	Memory	Additional Disk Space	Remarks
OS/2 Warp 4.0	24 MB	10 MB + DCE	
OS/2 Warp 3.0	24 MB	10 MB + DCE	requires FixPack 21+
NT 4.0	24 MB	20 MB	requires ServicePack 3
Windows 95	16 MB	30 MB	

3.5 File System Layout and Space Requirements

Before the GSO software is installed, one should create the necessary file systems to separate the different data types, which are:

- Software packages with binaries, libraries and documentation
- Persistent data store to support the DCE and GSO services, for example the DCE registry and the CDS (Cell Directory Service) clearinghouse information
- Dynamically produced data, for example files that contain logging information and program core files

The separation serves two purposes. First, it provides fault isolation in the event that when other processes filled up their filesystems, the GSO servers are still able to provide services. Second, the separation supports high availability configurations, for example High Availability Cluster Multi-Processing/6000 (HACMP/6000) for AIX configurations, where the configuration data and the persistent data has to be shared between different machines.

Table 10 contains a list of recommended file systems that one should add to AIX GSO server machines prior to installing the GSO software.

Table 10. File System Sizes for GSO Servers

Filesystem	Recommended Size	Remarks
/var/dce (AIX) or /opt/dcelocal/var (Solaris)	110 - 140 MB	This file system is mainly used for some configuration files and for the log files. See also Table 6.
/var/dce/security (AIX) or /opt/dcelocal/var/security (Solaris)	30 MB + 10 KB/user	This is where the DCE Security server stores the registry, credentials and local data. This also includes the GSO user data.
/var/dce/directory (AIX) or /opt/dcelocal/var/directory (Solaris)	30 MB	This is where the CDS server stores the clearinghouse files, which contain this server's portion of the namespace and local data.
/var/gso (AIX) or /opt/dcelocal/var/gso (Solaris)	8 MB	This is where the GSO server stores its configuration and log files.

Filesystem	Recommended Size	Remarks
/krb5	4 MB	This is where the Kerberos information, including local keytab files, is stored. If HACMP/6000 is not used, this directory may be part of the root file system.
/etc/dce	4 MB	This is where DCE configuration information is stored. If HACMP/6000 is not used, this directory may be part of the root file system.

Note

During normal operations, DCE credentials are stored as files in the /var/dce/security/creds directory (IBM AIX). These credential files may accumulate over time and can fill up the file system unless they are cleaned up regularly. To clean up expired credential files, use the `rmxcred` command. On AIX, this can be done by creating the following entry in root's crontab file, which removes expired credential files at midnight every night:

```
0 0 * * * /usr/bin/rmxcred >/dev/console 2>&1
```

Of course, this is only necessary if there is no other tool installed that performs this function.

3.6 Establishing Cell Names

In order to configure your GSO cell, you need to decide on a cell name. Choosing an appropriate cell name is important for the following reasons:

- Cells participating in the global namespace must have unique names that differentiate them from cells in other organizations.
- A uniquely identified cell name is critical to the operation of the security service. The name is the basis for authentication in your cell.
- Global cell names need to adhere to certain format standards.
- Changing a global name cannot easily be accomplished; therefore, do not chose a temporary cell name with a later change in mind.

If you plan to create a private cell or group of cells, and do not intend to interoperate with cells outside of your organization, you do not need to obtain

a globally unique cell name. However, for your cell to communicate with cells outside of your organization, you need to obtain a globally unique cell name from the GDS or DNS naming authorities. The registration of your cell name must be completed before you can begin to configure the cell namespace to avoid later changes or reconfiguration. Remember, even if you do not initially use a global directory service to communicate with other cells, you may want to do so in the future.

For example, if your organization already owns a dedicated DNS domain, you may pick a cell name for GSO as:

<gsoprefix>.<your domain name>

3.7 Tivoli Integration

The Global Sign-On for Multiplatforms, Version 2.0 product provides an integration into the Tivoli systems management product family. There is a Tivoli Plus module for GSO that handles deployment and the availability management of GSO servers and GSO clients. Additionally, there is a Tivoli GSO User Administration module that provides GSO user administration through a Tivoli endpoint for GSO. Please refer to Chapter 2, “Global Sign-On: The Macro View” on page 13, for more information about the integration into Tivoli.

3.7.1 Tivoli Management Regions versus GSO Cells

As far as the Tivoli Plus module for GSO is concerned, GSO servers are managed nodes, and GSO clients are either managed nodes or PC managed nodes.

GSO replica servers and GSO clients may be spread over separate Tivoli Management Regions. However, the GSO master server and also the GSO endpoint have to belong to one Tivoli Management Region (TMR). Because of this, the implementation of the Tivoli Management Regions can be done independently of the GSO cell design, or vice versa.

Figure 13 shows, as an example, the relationship of Tivoli Management Regions and GSO cells in a large Tivoli and GSO environment with two GSO cells and three Tivoli Management Regions. Both GSO cells in this example are managed from within one TMR. The GSO replica servers are distributed over several Tivoli Management Regions. However, the GSO master servers, and therefore the GSO cell endpoints, have to reside in the Tivoli Management Region from within which they are being managed.

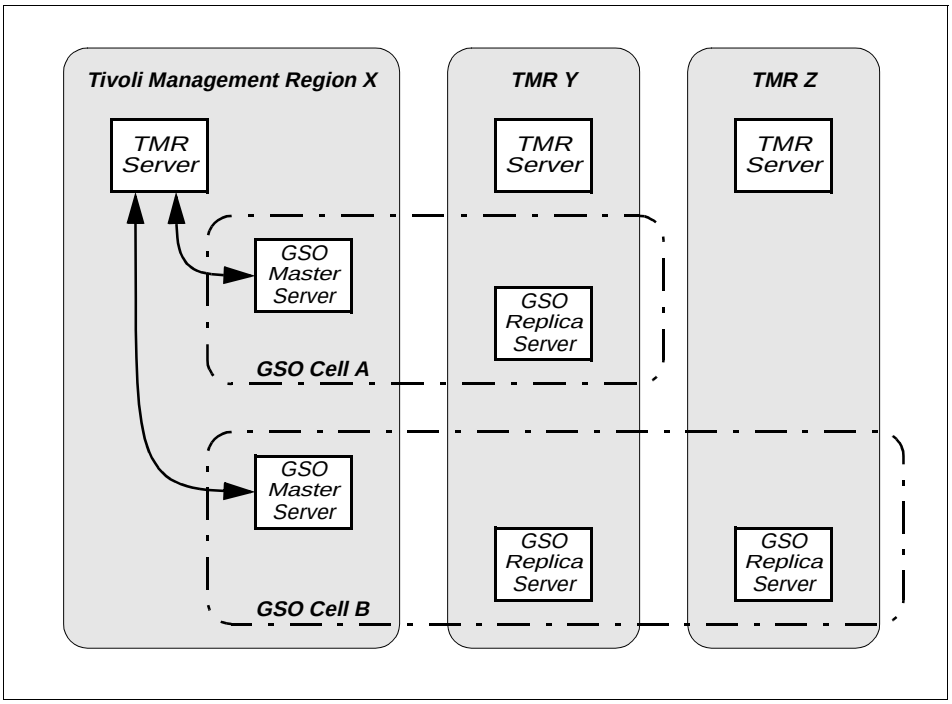


Figure 13. Interactions between GSO Cells and Tivoli Management Regions

GSO clients in Figure 13 can be anywhere, inside or outside any TMR, but they always belong to one (and only one) particular GSO cell.

3.7.2 The Tivoli Management Environment

Depending on the system and target environment, there exist many options on how the Tivoli setup can satisfy system administrators' needs.

This section presents an example for setting up the Tivoli Management Environment (TME) for the management of a GSO infrastructure that separates the responsibilities of different administrators. You should also read section 3.9.3, "Tivoli GSO Admin Security" on page 56, for additional information. Several policy regions and administrators are added in this sample scenario, as shown in Figure 14. In an actual implementation, the names of these object should conform to the established naming convention within the TME and target environment.

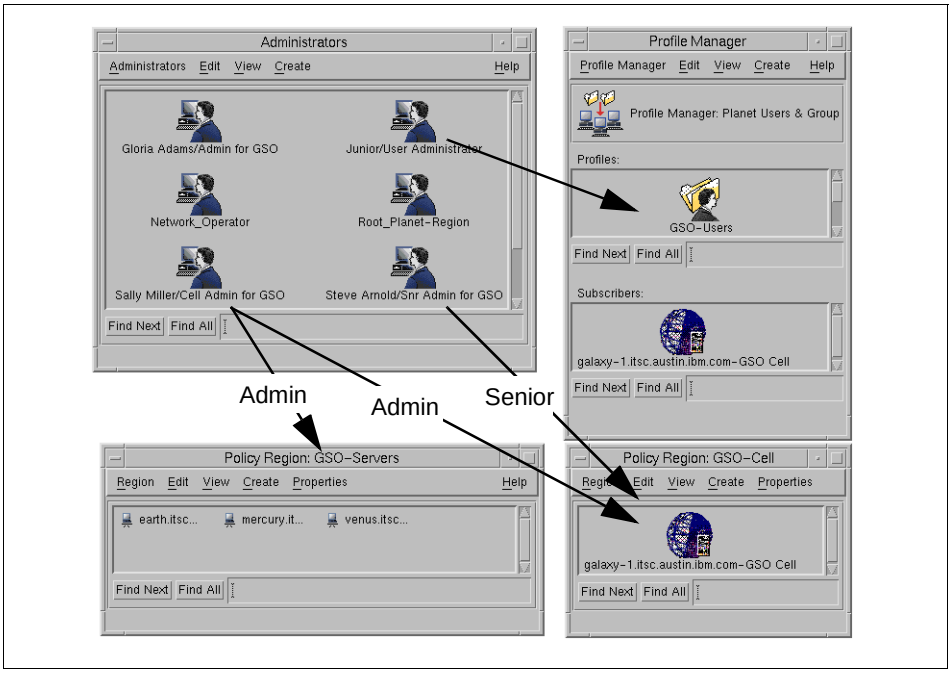


Figure 14. Tivoli Administrators and Their Authorization Roles

In this example, the following Tivoli policy regions are added to the TME:

GSO-Cell	Contains the cell object and manages **GSOCell** resources. If more than one GSO cell has to be managed, one or more policy regions or subregions can be created, or all cells can be controlled by one policy region.
GSO-Servers	Contains the managed nodes that correspond to the GSO server machines and manages **ManagedNode** resources. This policy region could further be divided up into GSO security server and GSO database server machines.
GSO-Monitoring	Contains the Tivoli Indicator Collection for monitoring GSO server resources.

Additional policy regions could be added, for example, for grouping the GSO clients, for software distribution purposes, or if more than one GSO cell is to be managed. Most likely, GSO is installed into an existing Tivoli environment, and there are conventions already in place for the organization of policy regions and policy managers.

In addition to the GSO-related policy regions described above, different administrator roles could be defined in an actual installation. Once the policy regions and administrator roles have been defined and implemented, cell objects (one per GSO cell) can be defined and profile managers can be put in place that contain the actual manageable objects.

The following is a list of administrator roles that might be implemented in an actual GSO environment:

Tivoli Senior Administrator

This is actually not a GSO administrator, but rather a Tivoli administrator. This senior role is required for configuration and policy tasks, such as creating a GSO administrator, creating policy regions and assigning policies.

The Tivoli Senior Administrator is also required to create the GSO cell object and assign it to a policy region. This can be done either by using the Tivoli desktop GUI or by using the command line interface, as in the following example:

```
wgsochgendpt -n "<gsocellname>GSO Cell" -r GSO-Cell
```

GSO Senior Administrator

The GSO Senior Administrator has the following authorization roles assigned:

- *senior*, *user* in GSO-Cell policy region
- *admin* for the user profile manager

The GSO Senior Administrator is responsible for:

- Setting the GSO cell password
- Subscribing the GSO cell to a profile manager
- Passticket management (adding, deleting, editing, populating)
- Setup and update target type management
- Add, delete, edit, and view cell target group(s)
- Apply a generic target to user profiles

Moreover, because of the authorization roles, the GSO Senior Administrator can also manage user accounts.

GSO Cell Administrator

The GSO Cell Administrator has the following authorization roles assigned:

- *admin*, *user* in GSO-Cell policy region
- *admin*, *user* in the GSO-Servers policy region

The GSO Cell Administrator is responsible for:

- Removing machines from cell
- Backup and restoring cell
- Setting password policies
- Synchronizing replicas
- Moving the master server
- Recovering the replica server

Moreover, because of the authorization roles, the GSO Cell Administrator can also act as the DCE cell_admin principal and, therefore, perform almost any task that is listed for the GSO User Administrator. However, the GSO Cell Administrator cannot subscribe the cell to a user profile manager without the corresponding *admin* authorization role for the profile manager. Additionally, the GSO Cell Administrator can enable or disable the event adapter and start or stop the GSO server(s).

GSO Systems Manager

The GSO Systems Manager has the following authorization roles assigned:

- *admin*, *user* in GSO-Monitoring policy region
- *admin*, *user* in the GSO-Servers policy region

The GSO Systems Manager is responsible for monitoring and maintaining system resources on the GSO server machines.

GSO User Administrator

GSO User Administrators do not differ from other user administrators within the TME. They do not need to have access to the GSO cell object; hence they need not have any authorization role in the GSO-Cell policy region.

The GSO User Administrator has the following authorization roles assigned:

- *admin*, *user* for user management profile manager

3.8 Identifying the Targets

The Global Sign-On for Multiplatforms, Version 2.0 product supports several standard targets, where the integration is supplied with the product. Moreover, it is possible to integrate your own targets into the GSO environment. In respect to planning for targets, there are four categories of targets:

Database Targets

To enable database servers to become target systems, it is necessary to install and configure the GSO database add-on code on these servers.

During this configuration process, the database servers become DCE clients. The configuration of the GSO database server core requires administration and configuration skills for the targeted relational database system.

For the DB2 and Informix database management systems, the RDBMS implementation requires that every GSO user who will connect to the database server have a corresponding operating system account on the database server machine. Tivoli User Administration can perfectly be used for this purpose.

Passticket Targets

The passticket support in GSO makes it possible to move the authentication of a mainframe application user ID from RACF to the GSO authentication function, hence delegating security responsibilities. Before the passticket function can be used, certain configuration activities on the RACF system have to take place, as explained, for example, in the *OS/390 V2R5.0 Security Server (RACF) Security Administrator's Guide*, SC28-1915. The RACF configuration is separate from the Global Sign-On for Multiplatforms, Version 2.0 product. For each application that users can gain access to with the passticket, at least one RACF profile in the PTKTDATA class has to be created on the OS/390 system. The profile associates a secret secured sign-on application key with a particular application on a particular system. The profiles can be created so they apply to:

- All users who need access to the application
- A specific RACF group of users who need access to the application
- A specific RACF user, when connected to a specific RACF group
- A specific RACF user

Other Standard Targets

For most targets, the use of GSO as the primary authentication method is transparent to the target systems. These targets can be used as before, and no, or only minor, GSO-related planning and customization has to be done. No GSO-specific configuration is needed for:

- Novell NetWare
- Windows NT 4.0 Server

Table 11 contains a list of targets supplied with the Global Sign-On for Multiplatforms, Version 2.0 product which require GSO-specific configuration changes, as further explained in Chapter 6, "Defining Targets" on page 147,

6.5, “Implementing SnareWorks” on page 186, and in the *TME 10 GSO User Administration* manual shipped with the GSO product.

Table 11. Standard Targets Requiring GSO-Specific Configuration

Target	Configuration Tasks
3270/5250 Emulator	Logon script files have to be created. The EHLLAPI function of the emulator has to be enabled.
AS/400	A command file for mapping local drives to AS/400 resources has to be created.
Lotus Notes	The EXTMGR_ADDINS variable in the notes.ini file has to be added or modified.
IBM OS/2 LAN Server	The WRKHEURISTICS keyword in the ibmlan.ini file has to be modified.
SnareWorks	DCE intercell trust has to be established between the GSO cell and the SnareWorks DCE cell.

Custom Developed Target Support

To cover targets not supplied by the IBM Global Sign-On for Multiplatforms, Version 2.0, product, GSO provides the interfaces to add your own targets. Please study the *GSO Programming Guide* (available on the product CD-ROM) and read section 6.7, “Adding New Targets to the GSO Framework” on page 199, in this book for more details.

3.9 Planning for Security

The implementation of the GSO environment should be in accordance with the security policy that applies to the target environment; that is, it should provide the right level of confidentiality, integrity and availability. Sensitive data, such as user passwords, should be protected from unauthorized disclosure; the data should be accurate and complete, and the service has to be supplied in compliance to service-level agreement. Once implemented, the security of the production GSO environment has to be monitored and the results checked against the security policy.

When deploying the Global Sign-On for Multiplatforms, Version 2.0 product, the following should be considered in respect to security:

- Users only authenticate with GSO. The authentication to the target system(s) is done automatically (or on user request) by GSO. Therefore, once a user is authenticated with GSO, he/she can log on to any of his/her targets without further authentication. Special attention must therefore be spent on locking the desktop when leaving the workplace.

- GSO uses the superior security services provided by DCE to store and retrieve user account information. The security of the GSO service is therefore very much driven by the security of the underlying DCE components, which must not be compromised.
- User account passwords are stored encrypted in the DCE registry.
- If the administration of the GSO environment is done through the Tivoli Management Environment, the security of the Tivoli components mainly drive the administration security.
- The use of the GSO passticket service delegates the primary responsibility of the authentication process from the authentication process of the target system, for example RACF, to the GSO environment.

The following areas should therefore specifically be addressed during the planning phase, as further outlined in the sections that follow:

- Installation and customization process – Prevent from unauthorized modifications
- Server security – Physical and logical access to server systems
- Administration security – To ensure proper administrative work
- Monitoring and auditing – To have a means of tracking security-related event

3.9.1 Installation and Customization Process

As with other security sensitive systems, it is necessary to ensure the integrity of the GSO server and client systems during the installation and customization process. For an unauthorized person, it should not be possible to add or modify software or hardware components on the GSO server and client systems. This could impact the security of the overall environment later on.

3.9.2 Server Security

The overall security, confidentiality, integrity, and availability can be compromised if an unauthorized person can log into a security server and obtain the GSO master key or other sensitive data. For this reason, the GSO master server and all GSO replica servers have be secured from unauthorized access. You will find information about securing your server system in the system documentation and other documentation about security or in the redbook *Protect and Survive Using IBM Firewall 3.1 for AIX,* SG24-2577, which contains valuable information on how to secure an AIX system.

3.9.3 Tivoli GSO Admin Security

The administration of the GSO environment, including the user account management, is one of the core areas where security has to be planned carefully. Having the appropriate permissions, a GSO administrator can modify account passwords and, potentially, authenticate to the users' targets. In the following sections , the different administration tasks and the required authorizations are listed. Using this information, together with the functions Tivoli provides, the scope and responsibilities of administration can be defined.

In respect to security, Tivoli GSO administration consists of three parts that need different authorizations to perform the corresponding task:

- GSO Cell Administration
- GSO Server Administration
- GSO User Administration

As part of the planning process, it has to be decided which part of the GSO administration should be assigned to which administrator group. The scope of administration—that is, the set of resources which an administrator controls—can be determined by Tivoli functions. The authentication of administrators to Tivoli and security aspects of the transport of messages within the Tivoli Framework are controlled by the Tivoli Framework and are outside of the Global Sign-On for Multiplatforms, Version 2.0 product. Therefore, it is assumed that Tivoli is implemented according to the security policy that applies to the targeted environment.

GSO Cell Management

The GSO cell management consists of functions that modify the GSO cell as an entity. These functions are provided through the Tivoli GSO Plus module.

Note

In order to perform most of the GSO cell management tasks, administrators need the most powerful authorization, meaning they can act as the DCE cell administrator and are able to control the GSO cell, add user accounts, change account passwords, and manage passtickets. The Tivoli role authorizations that accomplish this are *admin* and *senior* on the policy region that owns the GSO cell object (default: GSO Plus).

Table 12 lists cell management tasks together with the required authorization role in order to perform that task.

Table 12. Tivoli Authorization Roles for GSO Cell Management Tasks

Activity	Context	Required Role
Create and delete a GSO cell	Profile manager	senior
Query and modify the cell object, including query and change the GSO cell passwords	Policy Region that owns GSO cell object (default: GSO Plus)	admin
Subscribe the GSO cell to a profile manager	Subscribing profile manager	admin
	Policy region that owns the GSO cell object (default: GSO Plus)	admin
Passticket management (add, delete, edit, populate)	Policy Region that owns GSO cell object (default: GSO Plus)	senior
Set up and update target type management	Policy Region that owns GSO cell object (default: GSO Plus)	senior
Add, delete, edit, view cell target group	Policy Region that owns GSO cell object (default: GSO Plus)	senior
Apply a generic target to user profiles	Policy Region that owns GSO cell object (default: GSO Plus)	senior
Remove machine from Cell, Backup and restore Cell, set password policy	Policy Region that owns GSO cell object (default: GSO Plus)	admin
	Policy Region that contains GSO master server as a Managed Node	admin
Synchronize replicas, move master server, recover replica	Policy Region that owns GSO cell object (default: GSO Plus)	admin
	Policy Region that contains GSO server as a Managed Node	admin

In typical environments, as a rule of thumb, the *admin* or *senior* authorization roles for the policy region that contains the GSO cell object are granted only to a few administrators. These administrators set up the cell, which includes managing the passticket service and maintaining the Tivoli profiles used for user management.

The GSO servers contain sensitive data. Therefore, it may be appropriate to put all GSO server machines or only the GSO master server into a separate Policy Region.

GSO Server Management

The GSO server management includes functions that act on the servers that run the GSO services. These functions are provided through the Tivoli GSO Plus module.

Table 13. Tivoli Authorization Roles for GSO Server Management Tasks

Activity	Context	Required Role
Enable and disable event adapter	Policy Region which contains GSO master server as a Managed Node	admin
Start, stop server	Policy Region which contains GSO server as a Managed Node	admin

GSO User Management

The GSO user management consists of functions to create, delete, and modify GSO user account records within the Tivoli User Management module. These functions are provided as an add-on through the Tivoli GSO User Management module. As such, the security model for the GSO user management is based on the corresponding profile managers. The setup of these profiles and their subscribers is usually done by administrators who administer the GSO cell—that is, the administrators that are entitled to the *admin* role for the GSO cell object (see Table 12 on page 57).

To do the actual user account management, administrators essentially need the *admin* role for the corresponding profile manager, as listed in Table 14. It should be noted that the user account management functions are independent from the GSO cell and server administration and do not require the account administrators to be entitled to the *admin* or *senior* role on the policy region that contains the cell object.

Table 14. Tivoli Authorization Roles for GSO User Management Tasks

Activity	Context	Required Role
Add, delete, edit a user account	User profile	admin
Add, delete, edit targets for a user account	User profile	admin

Activity	Context	Required Role
Linking password to other attributes	User profile	admin

GSO Client Management

The GSO client management consists of functions for changing the client behavior, for example, to enable or disable integrated login through the Tivoli GSO Plus module. These management tasks require that the administrators possess the Tivoli authorization *admin* role for the Tivoli Policy Region that owns the GSO client. Table 15 summarizes these activities and roles.

Table 15. Tivoli Authorization Roles for GSO Client Management Tasks

Activity	Context	Required Role
Enable and disable integrated login	Policy Region that owns the GSO client	admin
Enable and disable Litronic Smartcard	Policy Region that owns the GSO client	admin

In addition to this, the configuration of GSO programs has to be done on the client. This can be achieved using the gso command line interface as documented in the GSO online command reference or through the graphical user interface. An authorization is needed to modify the GSO Program Database, %IBMGSOPATH%/config/pgm.db, on Windows NT. This can be achieved by using the NT administrator account. Moreover, in order to process the program configuration, it is necessary to be authenticated as an arbitrary GSO user. For large installations with similar operating system configurations, it may be appropriate to generate the GSO program databases centrally and to distribute these to the GSO clients.

3.9.4 Monitoring and Auditing

Once the GSO environment is implemented, the production environment has to be monitored, and the security-related events have to be audited. The Global Sign-On for Multiplatforms, Version 2.0 product comes with a monitoring module that hooks into the Tivoli environment (see also Chapter 7, “Managing GSO” on page 205). It has to be determined which GSO resources need to be monitored and which audit events are to be stored or forwarded to a security administrator.

The Tivoli GSO Plus module covers the monitoring of the GSO servers, such as disk, file system and file-size monitoring. Moreover, the underlying DCE infrastructure provides audit events for security-relevant activities with the

DCE security, time and audit servers. For example, using the DCE audit facility, it is possible to obtain audit events whenever account records and extended registry attributes are created or modified. Please refer to the *DCE Administration Guide - Core Components* (an online manual shipped with the DCE for AIX product) for more details.

Chapter 4. Installing GSO Servers

This chapter explains a step-by-step installation procedure for the GSO server, the GSO replica server(s) and the GSO database server(s). In addition to the installation procedure, various prerequisites and considerations are also explained which will enable you to take relative precautions in order to make the installation smoother.

4.1 Review: GSO Server, Replica Server(s) and Database Server(s)

Before moving on, it might be worthwhile to quickly review the different server types, as introduced and explained in 2.3, “GSO Servers” on page 20. They are: GSO (master) server, GSO replica server, and GSO database server. The first two are responsible for user authentication and secure storage of user configuration and logon information, such as user IDs and passwords. There must be a master server in every GSO cell since updates can only be done through a master server, but there can only be one master server in a GSO cell. Such updates include any kind of user administration (add, change, delete) through the use of the TME 10 GSO User Administration or password changes initiated by the users. The GSO (master) server is therefore a critical component for any update operations. It is often referred to simply as the GSO server.

GSO replica servers, if installed, provide an additional margin of availability because GSO clients can connect to either a GSO server or to any GSO replica server to access (read) user data. It is therefore highly recommended to have at least one GSO replica server installed in a GSO cell (see also 3.1, “The Environment” on page 34) because clients most often only do read accesses. GSO replica servers also help improve performance because client load can be distributed among multiple servers.

Any updates performed on a GSO server is automatically distributed (replicated) to the replica server(s); thus, no separate management or administrator action is necessary for the replica server(s) other than normal system monitoring and administration.

GSO database servers differ from the servers just explained above. A GSO database server does not store or maintain any user data, but it runs the server part of an RDBMS authentication system as part of GSO. Its sole function is to authenticate users to an RDBMS that is running on the same machine (which is not required, but recommended). GSO database server(s) are only required when GSO is being used for user authentication to an RDBMS.

4.2 Installation Prerequisites and Considerations

Before beginning the installation of GSO 2.0 on a server, there are some important considerations associated with the platform on which GSO is to be installed. It is also presumed that you will distribute and install the server software using the Software Distribution module of Tivoli for which some of the prerequisites and considerations that follow apply.

Note

For the latest and most up-to-date list of prerequisites, as well as known limitations and problems, you should always consult the release notes (usually a README file on the distribution media) of the product.

4.2.1 Common Prerequisites and Considerations

Mentioned below are some common considerations that must be made prior to installation. Some of the points are also useful in helping you plan your setup.

- If the GSO master server is installed on an IBM AIX system, any GSO replica (if created) must also be installed on AIX system(s).
- GSO servers and GSO database servers can be distributed only to Tivoli managed nodes.
- GSO clients and GSO database clients can be distributed to Tivoli managed nodes as well as to PC managed nodes.
- If the GSO master server is being installed on a Solaris or a Windows NT system, replicas (if created) can exist on Solaris, AIX or Windows NT.
- The GSO client system time and the GSO server time need to be synchronized within certain tolerances (normally 15 minutes). This is important for GSO authentication to work correctly. If there is a time difference of more than 15 minutes (default configuration) between the GSO clients and GSO servers, problems will occur.
- If a system that is planned to be a GSO server or replica server has more than one network interface (also called a multi-homed host), make sure that all interfaces can be reached by the clients. This includes, for example, an X.25 or an asynchronous SLIP interface line. By default, clients randomly select any of the available network interfaces when connecting to a GSO server, which may result in slow operation due to connection time-outs. Alternatively, network interfaces that are not available to clients should be disabled for GSO. This can be done by setting an environment variable in /etc/environment (UNIX). The following

example line in /etc/environment prevents an Ethernet interface from being used by GSO:

```
RPC_UNSUPPORTED_NETIFS=en0:et0
```

Note that this has no effect for other TCP/IP traffic over this interface.

- The GSO database server cannot reside on the same system where the GSO server is installed. It is preferable that the GSO database server is installed on the same system on which your application database (for example, Oracle, Sybase or DB2) resides. However, the GSO database server can be installed on a different system altogether. In that case, the client of the application database should also be loaded on the same system to initiate communication. This, however, affects the level of security since the security of transactions between the GSO database server and the application database server will depend on the application database server's security only.
- Review the installed TME packages on the systems where you plan to install the GSO server or the GSO database servers. This is to ensure that there is no prior installation of TME 10 GSO User Administration or GSO Plus on these systems. Use the Tivoli administration command

  ```
  wlsinst -ah
  ```

  to list installed products.

4.2.2 AIX Managed Nodes

Add the following line to the /etc/Tivoli/oserv.rc file:

```
ulimit -d unlimited
```

This line should be added before distribution of GSO software packages. It is recommended that it be added before the start of the installation. This file is called during installation of the system to start the Tivoli *oserv* daemon. By adding this line, the data segment for this process becomes unlimited. After this change is made, you will have to restart Tivoli for the changes to take effect.

You should also check the maximum number of processes per user, which defaults to 40. This number should at least be doubled.

4.2.3 Windows NT Managed Nodes

As a precaution, it is recommended not to run any other applications on a Windows NT managed node while distributing and installing the GSO file package. This is a general precaution for any software installation on Windows NT.

Before installation of the GSO server on the Windows NT computer, you must ensure that Tivoli is using the correct msvcrt40.dll file.

The following procedure will help you to ensure the correct file is being used:

From a command prompt, type the following commands:

```
cd \winnt\system32\drivers\etc\Tivoli
setup_env
copy %BINDIR%\mslib\msvcrt40.dll %DBDIR%
copy %BINDIR%\mslib\msvcrt40.dll %DBDIR%\bin
copy %BINDIR%\mslib\msvcrt40.dll %DBDIR%\tools
cacls %DBDIR%\msvcrt40.dll /e /g everyone:r
```

(`DBDIR` and `BINDIR` are environment variables for Tivoli set by the `setup_env` command.)

To operate properly, GSO requires access to the hardware system address of the managed nodes. In Windows NT, this is given by the NetBEUI protocol. Ensure NetBIOS is enabled on Windows NT. The procedure for checking whether NetBIOS is enabled is as follows:

- Double-click on the **Network** icon in the **Control Panel** to launch the network properties dialog box.
- Ensure that there is a NetBIOS interface already configured under the **Services** option. If there is no interface, then add the NetBIOS interface.
- Now select the **Protocols** option, and the following screen (Figure 15) should appear.

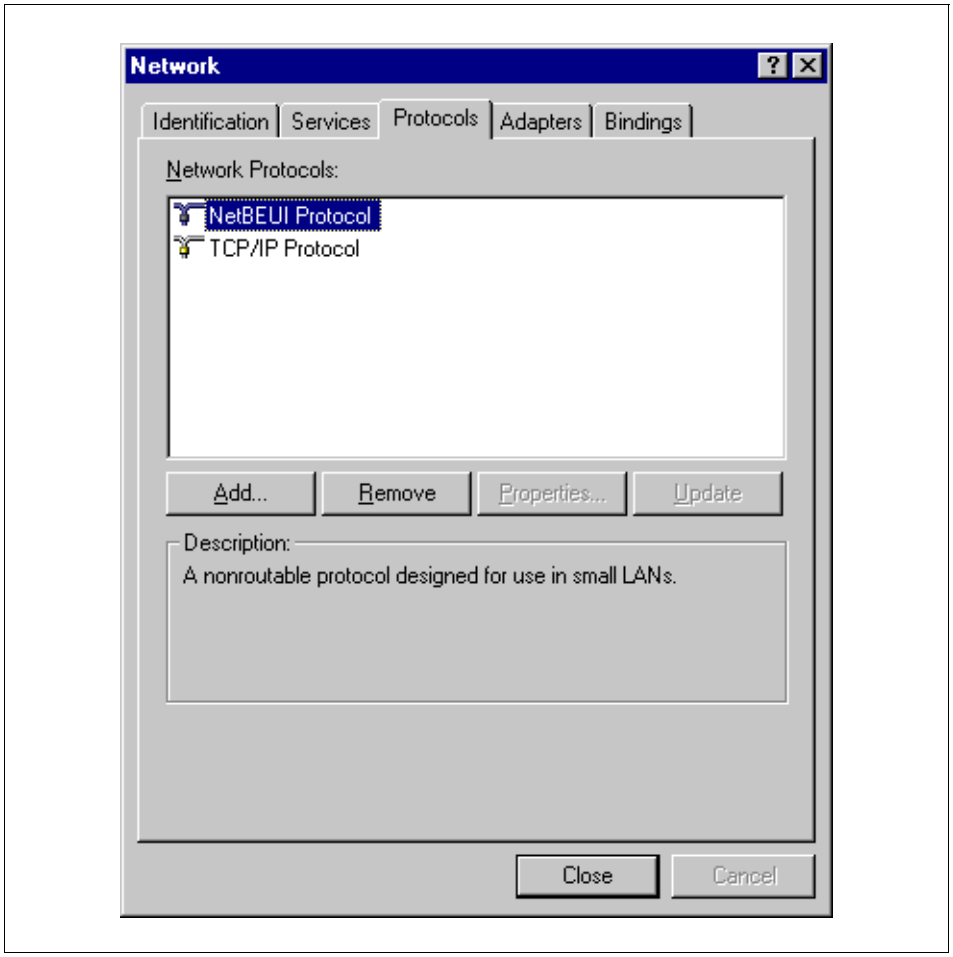


Figure 15. Checking the NetBEUI Protocol

If there are additional protocols configured in your network, you will see them here also. In case the NetBEUI protocol is not found in the list, you will have to add it using the **Add** button on that window.

- To ensure that NetBIOS is enabled, click on **Bindings**, and the following screen (Figure 16) should appear.

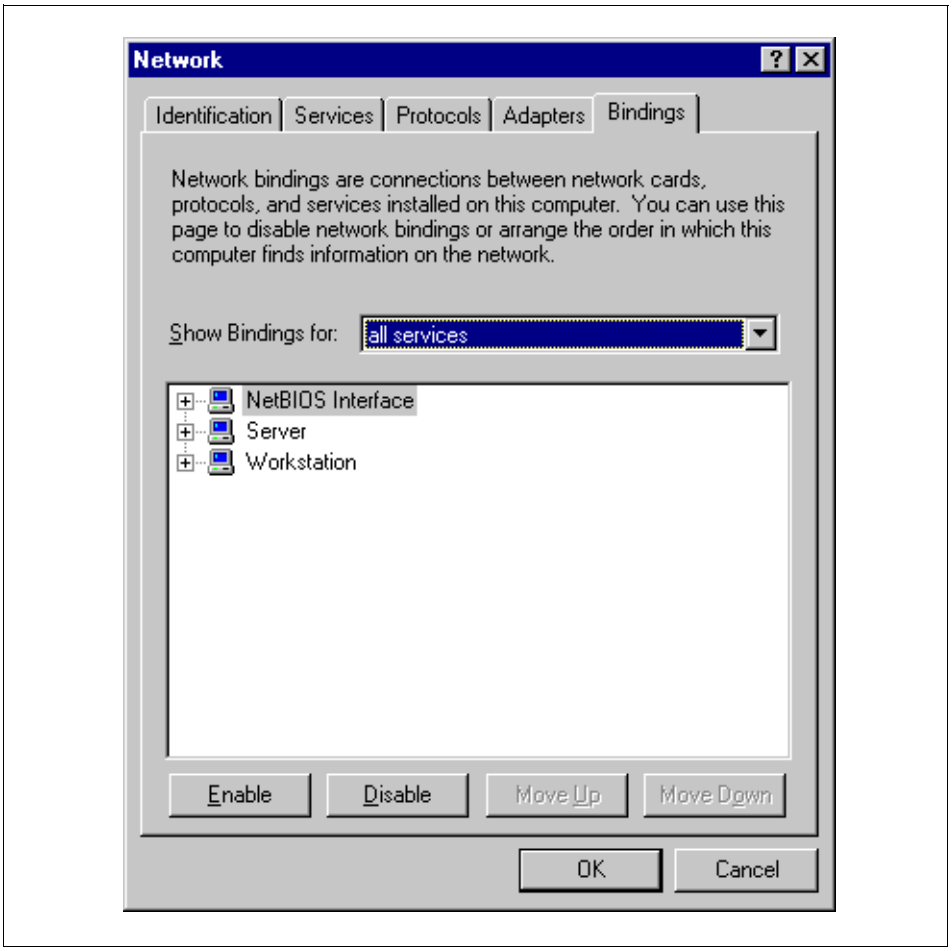


Figure 16. Checking the NetBIOS Interface

If the NetBIOS option is marked in red, you will have to enable the option by clicking on the **Enable** button.

4.2.4 Solaris Managed Nodes

Before a file package is distributed to any Solaris managed node, the following points should be considered:

- The Solaris node must have an /opt directory. If an /opt directory does not exist, create it as a subdirectory from the machine's root file system as a symbolic link to another directory in a file system on the machine or as a mount point for a partition on the machine.

- The computer must have a working network interface on which broadcasting is correctly configured.
- There should be an appropriate entry for Kerberos 5 in the Internet services database. This is either local in the /etc/services files or in a Network Information Services (NIS) service map. In the first case, the /etc/services file on your local computer must contain the entry mentioned below. In the second case, the same entry should be available in the services map file on the NIS master. The entry is:

  ```
  kerberos5 88/udp kdc
  ```

- The kernel of the computer must have shared memory enabled, and it must use semaphores. This is necessary because GSO uses shared memory. To verify whether these facilities are enabled, ensure that the following files exist:

  /kernel/sys/semsys
  /kernel/sys/shmsys

  You must also ensure that the following entries exist in the /etc/name_to_sysnum file:

  ```
  semsys number
  shmsys number
  ```

  where `number` is a system call number. In order to check for this, type the following at the command prompt:

  ```
  /bin/egrep 'semsys|shmsys' /etc/name_to_sysnum
  ```

- The /etc/group file should include an entry for the group `bin`.

4.2.5 PC Managed Nodes

Following are some considerations for Windows 95 PC managed nodes.

- The program environment space for the command processor needs to be increased in Windows 95-based systems. To ensure this, the following line should be added to the config.sys file:

  ```
  shell=c:\command.com /P /E:16384
  ```

- Windows 95 and Windows NT computers need to be rebooted after distribution of the GSO file packages.
- No configuration tasks can be performed until a user signs on to the workstation, which completes the install process. So ensure that a user signs on to the workstation.
- Only managed nodes are capable of returning messages to the distribution log file on the TMR server. Therefore, when a distribution fails,

the only indication that will appear will be a non-zero code (refer to Table 16 on page 68). However, the GSO distribution process creates a local log file named gsocinst.log on the PC managed node. This log file is created after distribution of the file package and is located by default in C:\Tivoli\tmeagent\Win95 in Windows 95 or in C:\Tivoli\tmeagent\Win32 or C:\var\spool\tmp in Windows NT. To check for the location of the file, do the following:

- Click on **Find** and select **Files or Folders** on the Windows **Start** bar.
- The following window (Figure 17) will pop up. Type in the file name, and select any appropriate drive to search on, as shown.

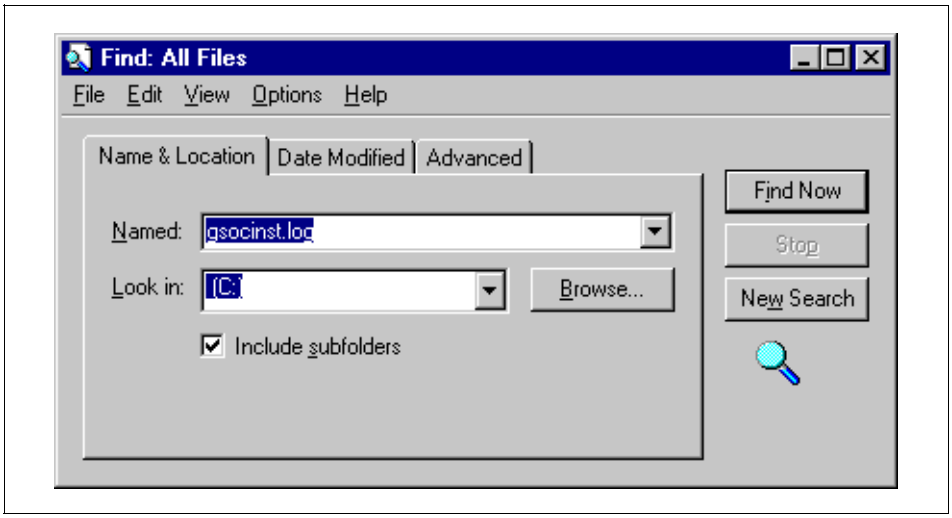


Figure 17. Window for Finding Files in the System

4.2.6 Distribution Return Codes

The distribution log file on the TMR server may contain error messages and list return codes. Table 16 lists these return codes with a brief explanation. Additional information may be logged in the local log file on the managed node as explained in the previous section.

Table 16. Distribution Return Code Table

Return Code	Explanation
2	All the required files for the file package were not transferred successfully.
3	Installation failed. Setup returned a non-zero return code.

Return Code	Explanation
4	Installation failed. The setup log could not be found.
5	Installation failed. Setup existed abnormally. The most likely causes of this error are: You are attempting to install a file package after previous removal of the same, and the workstation needs to be restarted. You are attempting to reinstall or upgrade GSO, but a GSO application is active. Users must end all GSO applications before you can redistribute a file package to that workstation. The Tivoli agent is not running with Windows NT System Administrator privileges.
6	Removal of the file package failed. The most likely cause of this error is that the Tivoli Agent is not running with Windows NT system administrator privileges.
7	GSO prerequisites are not installed. The GSO Client Version 2.0 must be installed before you can install the file package.
8	The staging drive or the installation drive does not contain enough space to install the file package.
9	Installation of the database client failed because the database type could not be determined. The most likely cause of this error is that the workstation does not contain enough environment space.
a	The file package cannot be installed. The file package was distributed to the wrong operating system type.

4.3 Installing GSO Tivoli Modules

GSO 2.0 server software is installed and configured using the Tivoli Software Distribution. Exceptions to this are the TME 10 GSO User Administration and the GSO Plus modules. They are extensions to the Tivoli Framework and are installed using the native Tivoli installation procedures, just as any other Tivoli application. The GSO Plus module is actually a prerequisite that must be installed in order to be able to use Tivoli to create and distribute GSO file packages.

The tables that follow give you an overview of the installation process. The detailed procedures follow in the subsequent sections. Table 17 summarizes the preparation steps.

Table 17. Steps Taken Prior to Installation

Step	Installation Step	Additional Information
1	Check the Tivoli Framework prerequisites.	For instructions, see the appropriate Tivoli documentation.
2	Back up your TMR server.	For instructions, see the appropriate Tivoli documentation.
3	Check the GSO operating system requirements.	For a list of supported operating systems and requirements, see 2.4, “Supported Platforms”, and 3.1.1, “Minimal Configuration”.
4	Review the GSO software and hardware prerequisites.	For details regarding the other considerations and prerequisites prior to installation, see 3.5, “File System Layout and Space Requirements”, and 4.2, “Installation Prerequisites and Considerations”.

Table 18 lists the tasks necessary to install the TME 10 GSO User Administration and the GSO Plus Module.

Table 18. Installing GSO Plus Module and Creation of Related File Packages

Step	Installation Step	Additional Information
1	Install the TME 10 GSO User Administration on the TMR server.	Details of the installation procedure are available in 4.3.1, “Installing the TME 10 GSO User Administration”.
2	Install GSO Plus on the TMR server.	For details, see the 4.3.2, “Installing GSO Plus”, and refer to the installation guide in the GSO product documentation.
3	Back up your TMR server again.	For instructions, see the Tivoli Framework Planning and Installation Guide.

4	Install GSO Plus on the Tivoli Enterprise Console server.	For instructions, see 4.3.2, “Installing GSO Plus” and GSO installation product documentation.
5	Set up the Tivoli Enterprise console.	For details, see the Tivoli and GSO Installation product documentation.

The steps listed in the tables above are explained in the sections that follow.

4.3.1 Installing the TME 10 GSO User Administration

The TME 10 GSO User Administration is used by Tivoli for adding users to the GSO cell after the installation of GSO is complete. It extends the functionality of the Tivoli User Administration by adding features specific to GSO (2.7.2, “User Administration” on page 31). The GSO server CD-ROM contains the software for TME 10 GSO User Administration in the /tme/gso/user directory. This file pack can be installed from any managed node in the Tivoli Management Region (TMR).

Following are the steps to be followed for the installation of the TME 10 GSO User Administration. It has to be installed on the TMR server, on the GSO (master) server, and on any managed node from which user administration is planned to be done.

1. Log in to the Tivoli desktop as an administrator with the `install_product` authorization role capability, and ensure that the GSO Server CD-ROM is inserted in the CD-ROM drive.
2. Select the **Install** -> **Install Product...** option from the **Desktop** pull-down menu. It is fairly common at this time that an error message pops up indicating that something is wrong with the installation media (which might be somewhat misleading). If you did get this error, click on **OK**, which brings up the File Browser window from where you can select the correct host and directory information to the GSO CD-ROM. The TME 10 GSO User Administration is located in the /tme/gso/user directory on the CD-ROM. Once host and directory information is entered correctly, the Install Product window appears, and TME 10 GSO User Administration should be listed in the Select Product to Install section.
3. Highlight **TME 10 GSO User Administration 2.0** from the Select Product to Install scrolling list by clicking on it.
4. In order to specify the computers where this product is to be installed, move the system names between the Clients to Install On and the Available clients scrolling lists by using the arrow buttons beside them.

5. Click on the **Install & Close** button to begin the installation process and close the Install Product window.

   This action pops up the Product Install dialog. It displays any problems that you might want to correct before the installation is completed. It will also show you the activities that are going to be performed while installation. In case you feel that you need to make some corrections prior to installation, click on **Cancel.** You will be returned back to the Install Product dialog.

6. Click the **Continue Install** button to begin the installation process.

   This dialog now displays a running log of the installation process as the installation proceeds. When the installation process is completed, this dialog returns a completion message.

   Note: Please read through the entire dialog since there might have been some errors during the installation. The dialog often gives a successful completion message at the end despite errors.

This completes the installation of TME 10 GSO User Administration on the managed nodes that were selected.

4.3.2 Installing GSO Plus

Following are the steps to be followed for the installation of the GSO Plus module. The GSO Plus module needs to be installed on the TMR server and on the planned GSO server(s). The GSO Plus module adds GSO-specific functionality to Tivoli as explained in 2.7, “Integration with Tivoli” on page 30 (except User Administration, which needs to be installed separately, as explained in the previous section).

1. Log in to the Tivoli desktop as an administrator with the `install_product` authorization role capability, and ensure that the TME GSO server CD-ROM is inserted in the CD-ROM drive.
2. Select the **Install** -> **Install Product...** option from the **Desktop** pull-down menu. You might get an error pop-up at this time that tells you that something is wrong with the installation media. If so, click on **OK** and select the correct host and directory name of the CD-ROM path in the File Browser window that comes up. The GSO Plus module is located in the /tme/gso/plus directory on the CD-ROM.

   After specifying the correct host and directory in the File Browser window, the GSO Plus module must show up on the Select Product window (Figure 18).

3. Highlight **GSO Plus, Version 2.0, Revision a** from the Select Product to Install scrolling list by clicking on it (the name or revision index in your installation might have changed slightly).

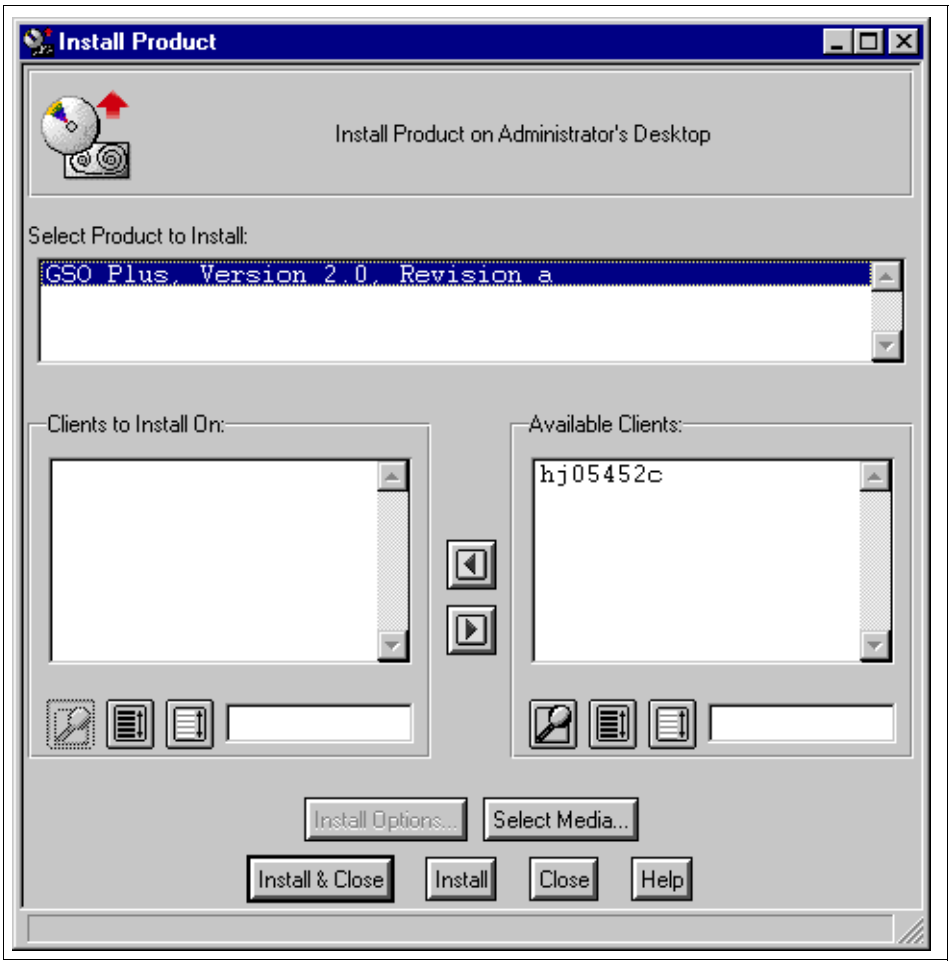


Figure 18. GSO Plus Selection of Managed Nodes for Installation

4. In order to specify the computers where GSO Plus is to be installed, move the system names between the **Available Clients** and the **Clients to Install On** scrolling lists by using the arrow buttons between them.
5. Click on the **Install & Close** button to begin the installation process and close the Install Product dialog.

   This action pops up the Product Install window (Figure 19). It displays any problems that you might want to correct before the installation takes place.

This window also displays a log of the activities that are performed during the installation process.

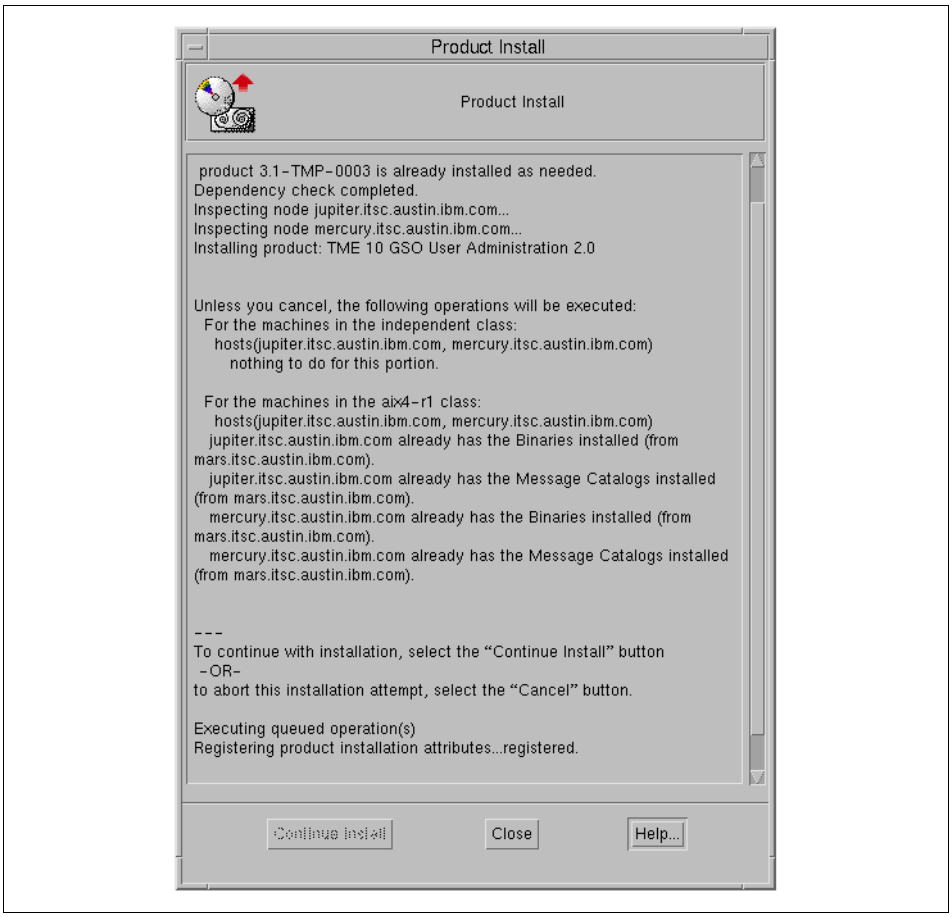


Figure 19. Product Install Dialog (Continue has already been clicked on)

6. Click the **Continue Install** button to begin the installation process, which is displayed on the Product Install dialog.

   This window now displays an updated status of the installation process as the installation proceeds. When the installation process is completed, this window concludes with a completion message.

   Note: You should always read through the entire output since there might be some errors during installation. The dialog may give a successful completion message at the end despite errors.

This completes the procedure for the GSO Plus installation. The completion can be verified by looking at the TME administrator desktop (Figure 20). A TivoliPlus icon has been added to the desktop (if not already added by another TivoliPlus product installation).

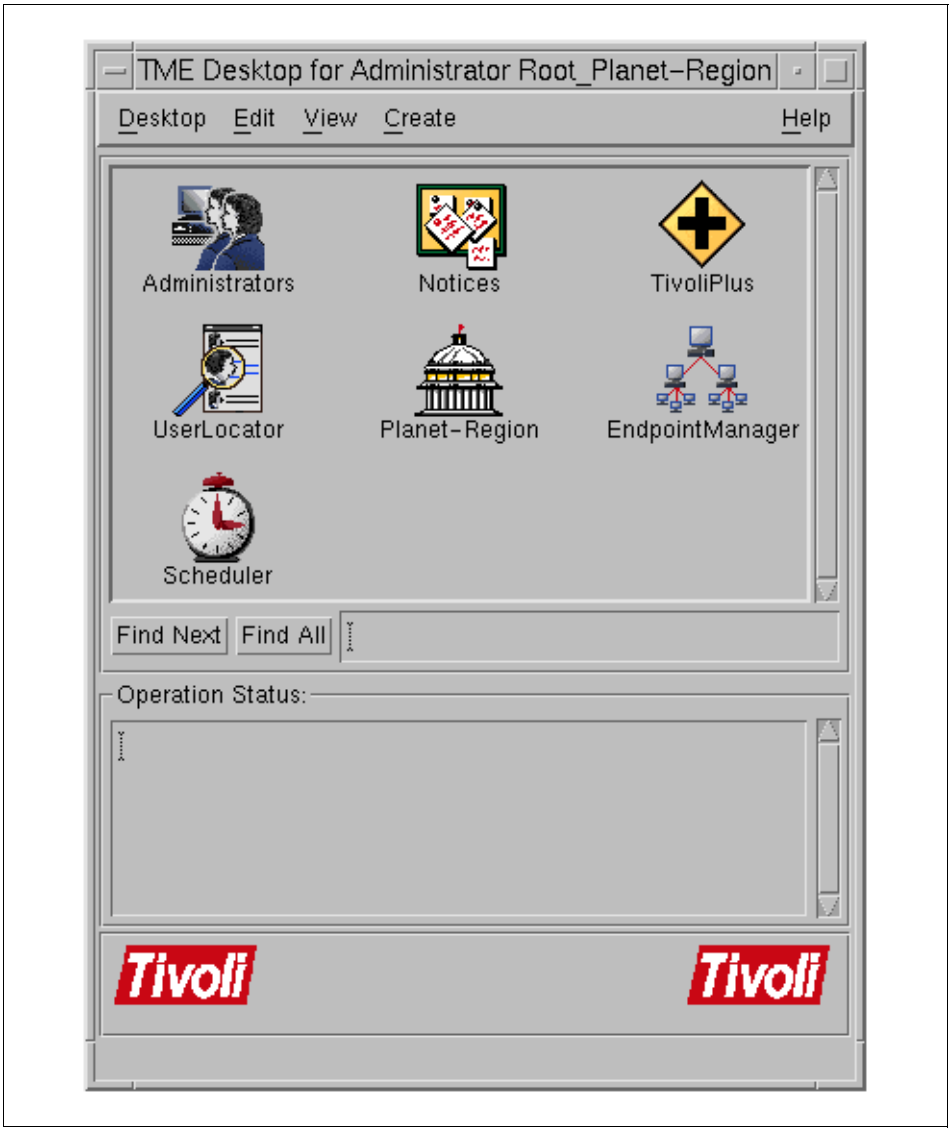


Figure 20. TivoliPlus Icon After Installation of GSO Plus

4.4 Installing Servers Using Tivoli Software Distribution

This section describes the distribution and installation of GSO servers using Tivoli-created file packages. These file packages need to be set up and distributed to the respective managed nodes according to their intended function. Table 19 gives you an overview of the installation steps.

Table 19. Steps for Distribution and Configuration of the GSO File Packages

Step	Installation Step	Additional Information
1	Create the file package for the GSO server.	For details of configuration, see 4.4.1.1, “GSO Server”, and GSO configuration product documentation.
2	Create the file package for the GSO database server (optional, only if GSO is used for user authentication to databases).	For details of configuration, see 4.4.1.2, “GSO Database Server” and GSO configuration product documentation.
3	Distribute the file package for the GSO server (master and any replica) to the managed nodes where GSO servers are installed.	For installation details, see 4.4.2.1, “GSO Server” and the GSO installation product documentation.
4	Distribute the file package for the GSO database server to the managed nodes where GSO database servers are installed.	For installation details, see 4.4.2.2, “GSO Database Server” and GSO installation product documentation.
5	Configure the GSO master server.	For configuration details, see 4.4.3.1, “GSO Master Server” and GSO configuration product documentation.
6	Configure a GSO replica server (optional, but recommended).	For details, see 4.4.3.2, “GSO Replica Server” and GSO configuration product documentation.
7	Configure the GSO database server on the managed node where the GSO database server is installed (optional, only if GSO is used for user authentication to databases).	For details, see 4.4.3.3, “GSO Database Server” and GSO configuration product documentation.

Note: GSO replica servers and database servers can be added or removed any time after the GSO master server is installed and configured.

The distribution of the software can be done on a multiplatform network. For example, if you have done the installation of GSO Plus on an AIX system, the GSO server software can be distributed to an AIX, Windows NT and/or Solaris system. Similarly, if you have done the installation of GSO Plus on a Windows NT system, the distribution of the GSO server file package can be done to any of the supported system platforms.

The installation of the GSO server is complete only after the distribution of the relative file packages to the respective systems. Do not remove the GSO server CD-ROM from the CD-ROM drive until the file packages are created. You must have senior administrative role authority to perform the activities mentioned below.

Note on CD-ROM Location

When setting up a file package for either type of GSO servers or clients, you must specify a host name and a path to the CD-ROM that contains the GSO server or client code, respectively.

It is important to know that creating a file package does not actually copy files from the CD-ROM to build up a large distribution package. The file package only contains a link to the CD-ROM path and the actual files will only be read from the CD-ROM when the distribution takes place (which might be days later). The files could also be copied from the CD-ROM to a hard drive, and that directory path would then be required for the file package.

4.4.1 Setting Up Server File Packages

A separate file package needs to be created for each operating system platform. Also the file packages that are created are different for a GSO server and a GSO database server. The procedure for setting up file packages for both server types are basically the same, but there are a few configuration differences as explained in the sections that follow.

4.4.1.1 GSO Server

The following steps will help you through the process of creating a file package for a GSO server. The steps mentioned below are the same for every operating system and have to be repeated for each operating system for which the file package needs to be created:

1. On the Tivoli desktop, double-click on the **TivoliPlus** icon.
2. In the upcoming TivoliPlus window, double-click on the **GSO Plus** icon.

3. From the window that now appears, double-click on the **Set Up GSO Server File Package** icon, which brings up the Set Up GSO Server File Package window (Figure 21).

Figure 21. Dialog for Setting Up the GSO Server File Package

4. In the File Package Name field, type a name for the package that you are creating. It is good practice to choose a name that describes the features of the package.

5. In the Source File section, type the host name and the directory path where the GSO server CD-ROM will be available for distribution of the package. On Windows NT, type the drive letter assigned to the CD-ROM.
6. In the Distribution Options section, select the Target Platform Operating System from the selection list. This is the operating system for which the file package is intended.
7. If you selected AIX or Solaris as the operating system, type in a directory path name for a directory on the destination system(s) that has enough free space to temporarily store the file package during the distribution and installation process (from observation, about 100 MB free space is well enough). The files in this path are deleted once the installation is completed.
8. If you selected Windows NT as the operating system, specify a distribution staging drive and directory and select the drive on which you wish to install the GSO server software.

   The distribution staging directory is used on the destination system to temporarily store files during the distribution and installation.

   It is necessary to restart Windows once the distribution of the file package is complete. You can do this manually or automatically by checking the **Restart Windows NT after distribution or removal** check box. If this option is checked, GSO will initiate a restart of the GSO server once the file package distribution is completed.
9. Click on **Set and Close** once this dialog is completed. This opens the Set Up GSO Server File Package Output dialog, as shown in Figure 22.
10. In the standard output section shown in this window, there should be a message stating successful completion of the creation process. The standard error output section should not indicate any errors (some messages might be misleading). If errors are noted, verify the errors and if required, redo the file package creation after correcting the cause of the errors.
11. The **Save to File...** button allows you to save the text contents of that window to a file. If you do not wish to save the dialog, click on the **Close** button.

Figure 22. GSO Server File Package Output Dialog

Timeout Errors?

Creating a server file package is a Tivoli job with an assigned timeout value. If the system being used is heavily loaded or slow for any other reason, creation of a file package can abort with a timeout error indication. This timeout value can be modified to prevent such errors from happening. Right-click on the **Set Up GSO Server File Package** icon in the GSO Plus collection window and select **Modify job...** from the pop-up menu. In the upcoming Edit Job dialog window, locate the Timeout value and increase it according to your needs (or specify 0 for indefinite).

This completes the creation of a file package for the GSO server for the operating system chosen. A file package icon for this package has been created in the Distribute GSO Server profile manager.

4.4.1.2 GSO Database Server

The following steps help you through the process of creating a file package for a GSO database server. The steps mentioned below are the same for every operating system platform and have to be repeated for each operating system for which the file package is to be created.

1. On the Tivoli desktop, double-click on the **TivoliPlus** icon.
2. In the upcoming TivoliPlus window, double-click on the **GSO Plus** icon.

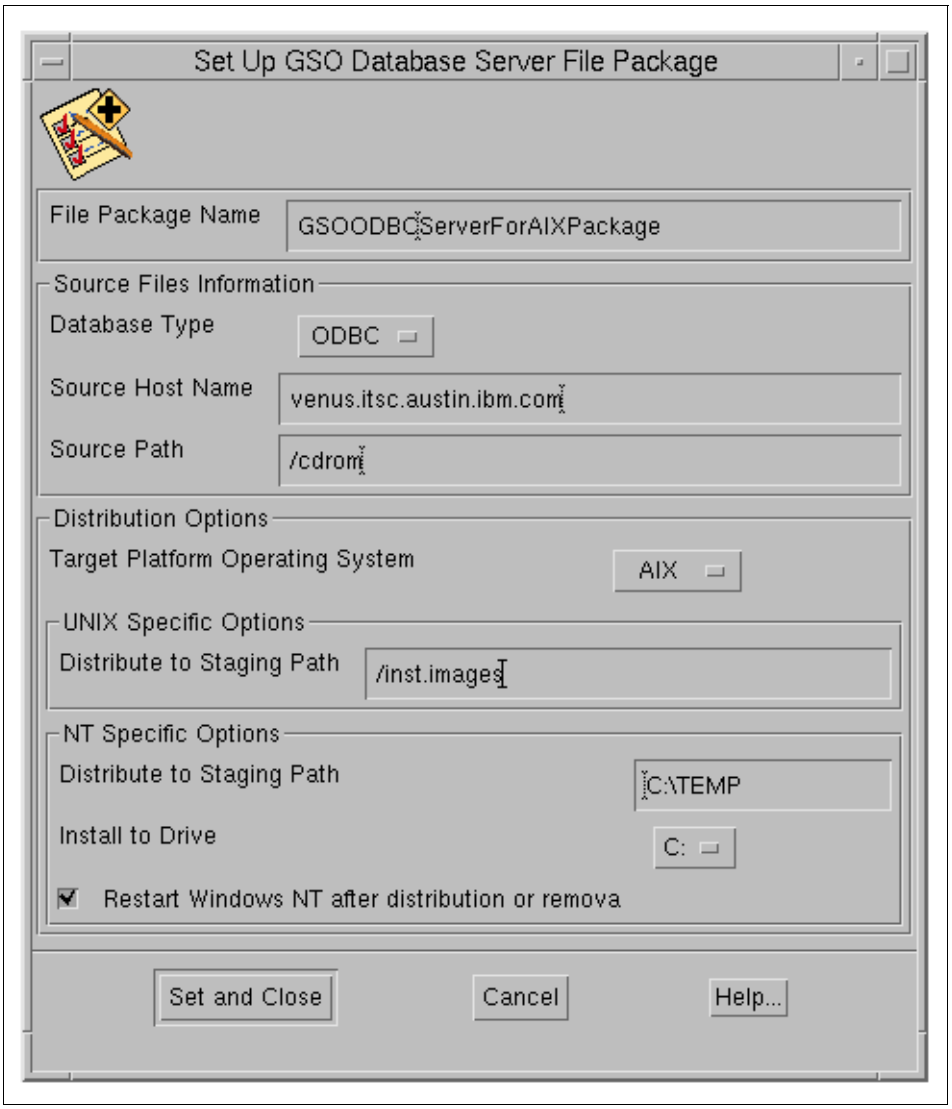


Figure 23. Dialog for Setting Up the File Package for GSO Database Server

3. From the window that now appears, double-click on the **Set Up GSO Database Server File Package** icon, which brings up the Set Up GSO Database Server File Package dialog window (Figure 23).
4. In the File Package Name field, type a meaningful name for the package that you are about to create.
5. From the Database Type selection list, select the type of database (or connection method) that you wish to configure the GSO database server for.
6. In the Source Host Name and Source Path fields, type the host name and the directory path where the GSO server CD-ROM will be available for distribution of the package. On Windows NT, type the drive letter assigned to the CD-ROM.
7. In the Distribution Options section, select the target platform operating system from the selection list for which this file package is intended.
8. If you selected AIX or Solaris as the operating system, type in a directory path name for a directory on the destination system(s) that has enough free space to temporarily store the file package during the distribution and installation process (from observation, about 100 MB free space is enough). The files in this path are deleted once the installation is completed.
9. If you selected Windows NT as the operating system, specify a distribution staging drive and directory and select the drive on which you wish to install the GSO database server software.

   The distribution staging directory is used on the destination system to temporarily store files during the distribution and installation.

   It is necessary to restart Windows once the distribution of the file package is complete. You can do this manually or automatically by checking the **Restart Windows NT after distribution or removal** checkbox. If this option is checked, GSO will initiate a restart of the GSO server once the file package distribution is completed.
10. Click on **Set and Close** once this dialog is completed. This will open the Set Up GSO Database Server File Package Output dialog (Figure 24).

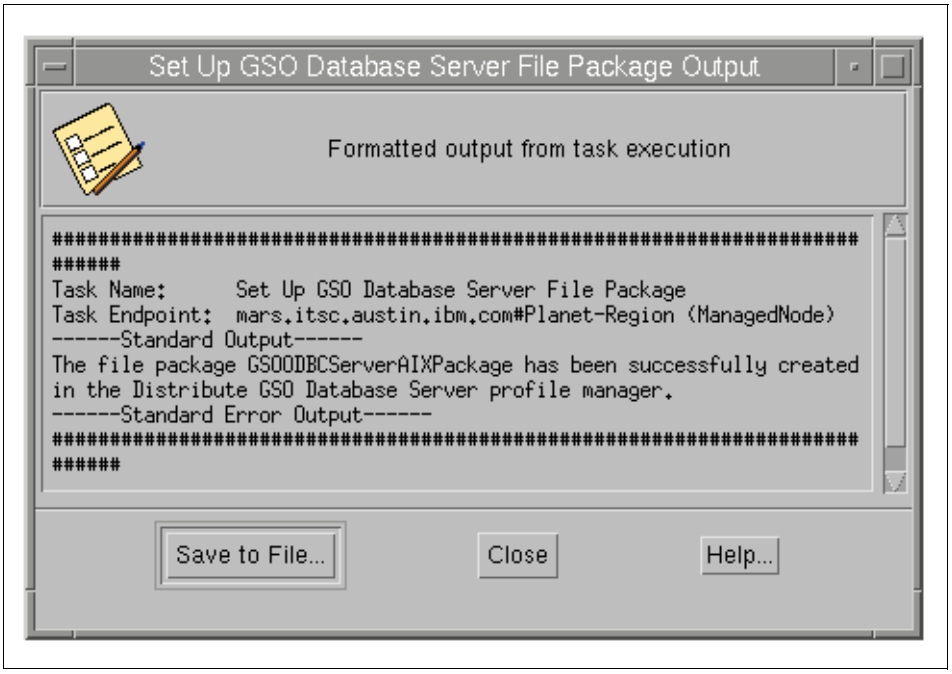


Figure 24. GSO Database Server File Package Output Dialog

11. The Standard Output section of this dialog should state successful completion of the package creation. The Standard Error Output section should not indicate any errors. If errors are noted, verify the errors and if required, redo the file package creation after the cause has been fixed. The Standard Error Output section sometimes notifies that it found the node of the installation device; however, this message can be ignored.
12. If you want to save the contents of the output, click on the **Save to File...** button. If you do not wish to save the messages, click on the **Close** button.

Note: If you get timeout errors, read the note on page 80.

This completes the creation of the file package for the GSO database server for the operating system and for the application database chosen. A file package icon for this database is created in the Distribute GSO Database Server profile manager.

4.4.2 Distribution of GSO Server File Packages

After packages have been created for GSO server(s) and GSO database server(s), they need to be distributed to the selected systems in order to get installed.

Note on Distribution

The steps described in this section describe a suggested way to distribute GSO server packages. The Tivoli environment supports other ways to implement package distribution that you might find are more adequate for your environment. There is even a command line interface that can be used for most (or even all) operations in Tivoli.

Whatever method you choose, make sure the requirements described here are met.

The distribution process creates a local log file on the distribution server (default file name is: <file package name>.log) in the Tivoli temp directory (for example /tmp on IBM AIX) during the process of file package distribution. (These defaults can be changed as shown in the following sections.) Please refer to 3.5, “File System Layout and Space Requirements” on page 46, for specific considerations and disk space requirements before distributing the file packages.

4.4.2.1 GSO Server

Follow the steps mentioned below to distribute the GSO server file package to the managed nodes that have been planned to become GSO servers.

1. From the GSO Plus window, double-click on the **Distribute GSO Server** icon.

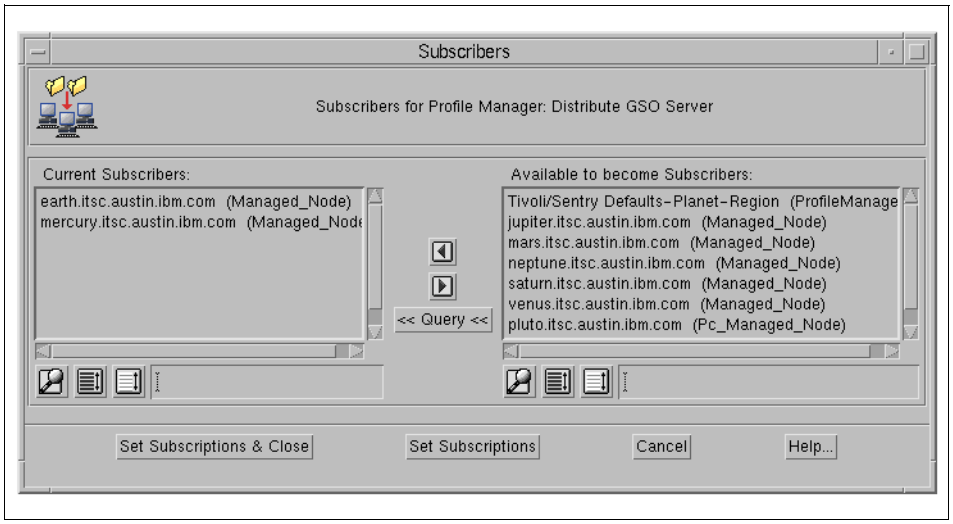


Figure 25. Defining Subscribers for Distribution of GSO Server File Package

2. The Subscribers dialog appears (Figure 25) with a list of available managed nodes in the Available to become Subscribers list on the right-hand side. In order to specify the computers where this file package has to be distributed, move the system names between the Available to become Subscribers and the Current Subscribers scrolling lists by highlighting them and using the arrow buttons in the middle.
3. Click on the **Set Subscriptions & Close button**. You will be returned back to the GSO Plus Window.
4. Click on the **Distribute GSO Server** icon with your right mouse button and select the option **Open...** from the pop-up menu.
5. This opens the Distribute GSO Server window (Figure 26), which will have the GSO server file package icon in it that was created in an earlier step. (The name of the icon will be the name of the file package for distribution of GSO server given by you as in 4.4.1.1, “GSO Server” on page 77.) In the example shown in Figure 26, the name of the icon is GSOServerForAIXPackage.

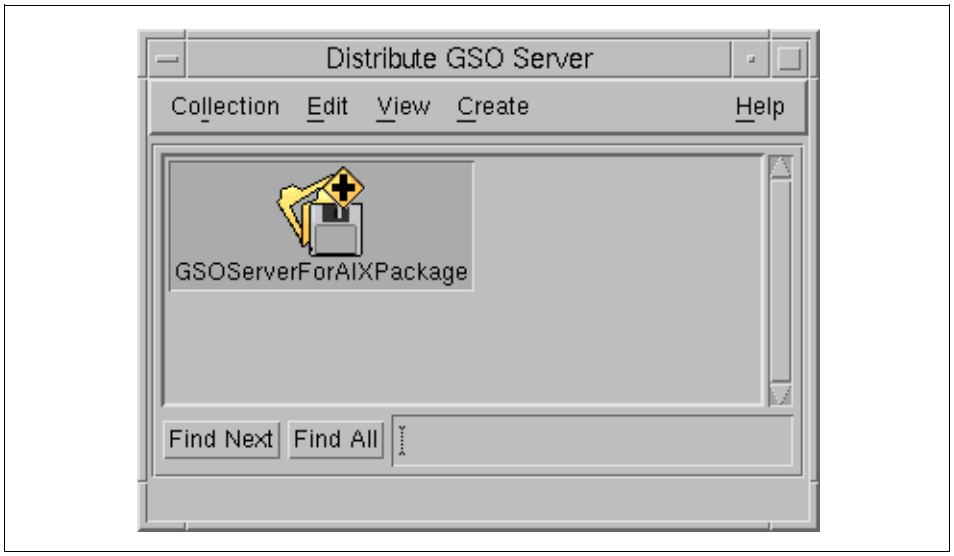


Figure 26. Window Displaying Icon for Distribution of GSO Server

6. Now click on this icon with the right button of your mouse and select **Open...** from the pop-up menu. This opens the File Package Properties dialog (Figure 27) that allows you to further specify properties for the distribution.
7. The Source Host section displays the host and directory from which the files are going to be installed. Do not change the directories and files listed as these are automatically chosen from the GSO CD-ROM based on the operating system for which the file package is created.

   Change other options as required. Normally, there is no need to change any option, but your environment might have other requirements.

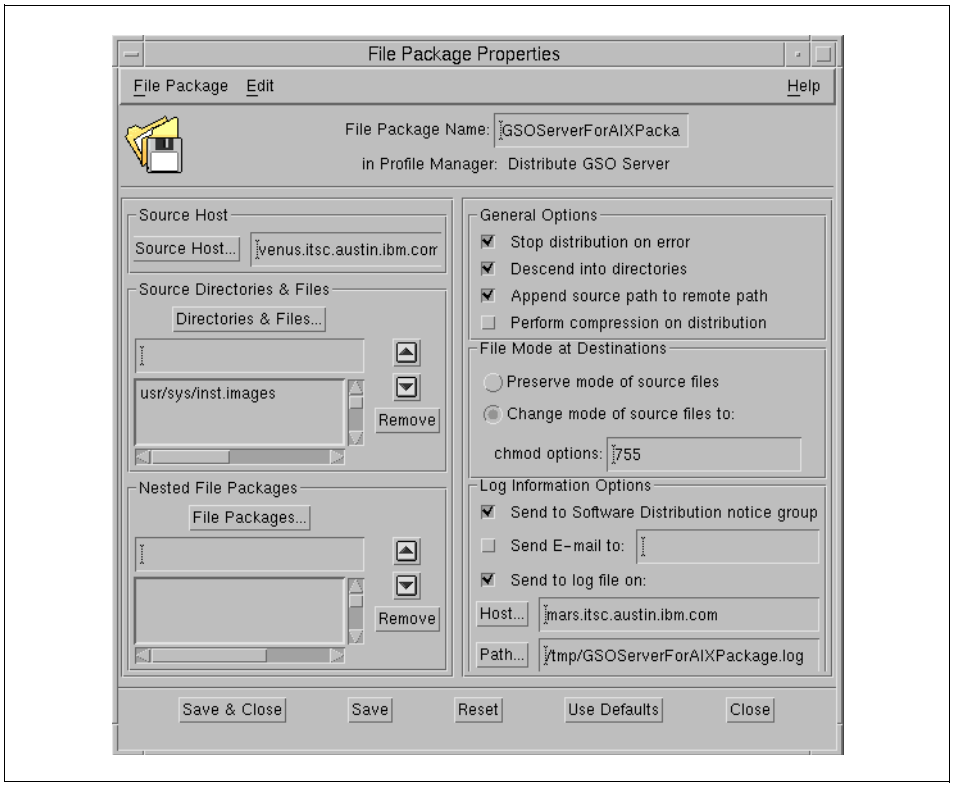


Figure 27. Dialog Displaying the File Package Properties for the GSO Server

Specify the log information options. Ensure that the Host and the Path under the Log Information Options are the locations where you want to store the output log file of this distribution. Change these details if you want the file in a different location. (The default path and file name on UNIX is /tmp/<file package name>.log, and on Windows NT it is c:\Tivoli\db\<hostname>.db\tmp\<file package name>.log.)

If you have changed any information in this dialog window, click on **Save** to save these changes.

8. Still on the File Package Properties window, select **Distribute...** from the File Package pull-down menu to open the Distribute File Package window, shown in Figure 28.

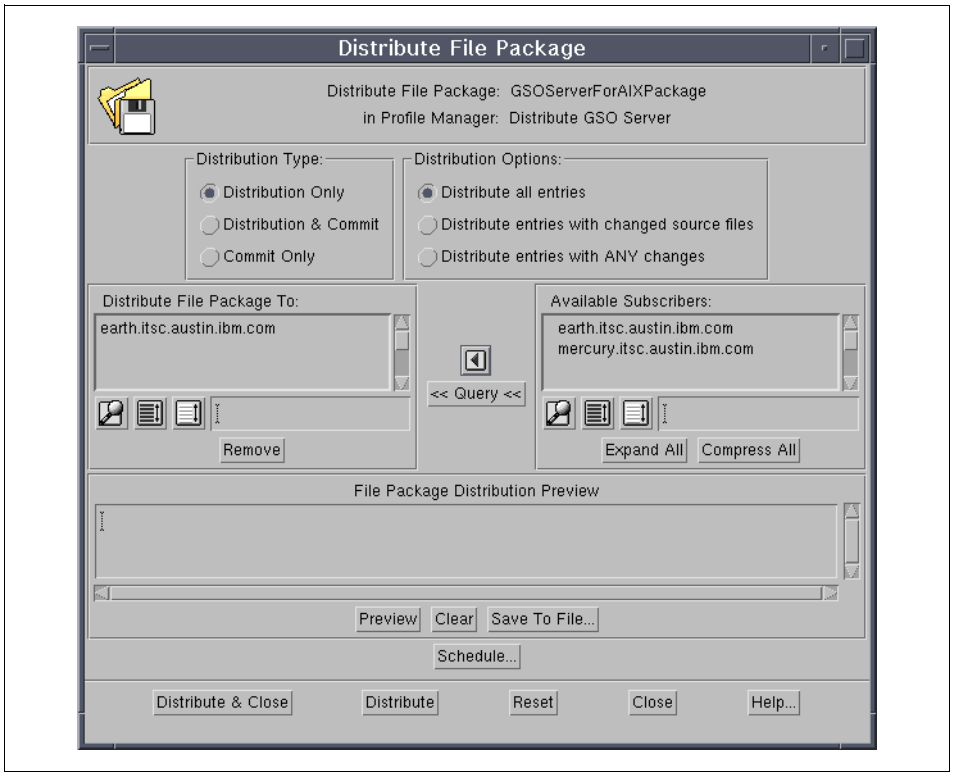


Figure 28. Distribute GSO Server File Package Dialog

9. Make sure that you move the correct system name(s) from the Available Subscribers list to the Distribute File Package To list. The GSO server file package will only be distributed to the system(s) that appear(s) in this list on the left-hand side. The Available Subscribers list contains those systems that have been selected in step 2 above.

   The other options normally do not need to be changed.

10. Click on **Distribute** (or **Distribute & Close**) to start the distribution process. Note that this panel also allows you to define a scheduled distribution—that is, an automatic distribution at a specified time in the future. Please read the Tivoli documentation for further information on this.

Distribution takes place in the background, and you cannot follow the process on the screen. After the file package is distributed, read through the <file package name>.log file and check for any errors. Common errors might be that there is not enough free space in the temporary staging directory or

that the GSO server CD-ROM is not inserted correctly in the drive at the time of the distribution.

Other Distribution Options

The file package icons in the GSO Plus and the Distribute GSO Server collection windows also have a **Distribute** option in their pop-up menus. You should not use this option since it starts a distribution process immediately for all defined file packages, not just for a single one.

4.4.2.2 GSO Database Server

Mentioned below are the steps to distribute the GSO database server to the managed nodes that are intended to become GSO database servers. (This step needs to be done only if there is a database server planned.) Please refer to 4.2, “Installation Prerequisites and Considerations”, and to 2.5, “Hardware Requirements”, prior to software distribution.

1. From the GSO Plus window, double-click the **Distribute GSO Database Server** icon.
2. The Subscribers dialog with a list of managed nodes opens (Figure 29). In order to specify the computers where this file package has to be distributed, move the system names between the Available to become Subscribers and the Current Subscribers scrolling lists by highlighting them and using the arrow buttons in the middle.

Figure 29. Defining Subscribers for Distribution of GSO Server File Package

3. After selection is done, click on the **Set Subscriptions & Close** button to return back to the GSO Plus window.
4. Click on the **Distribute GSO Database Server** icon with your right mouse button and select the **Open...** option from the pop-up menu.

   This opens the Distribute GSO Database Server window which contains a GSO database server file package icon for each such file package that has been created. The name of the icon corresponds to the name of the file package as specified when setting up the file package (see 4.4.1, “Setting Up Server File Packages” on page 77).
5. Now right-click on this icon and select **Open...** from the pop-up menu. This opens the File Package Properties dialog (Figure 30).

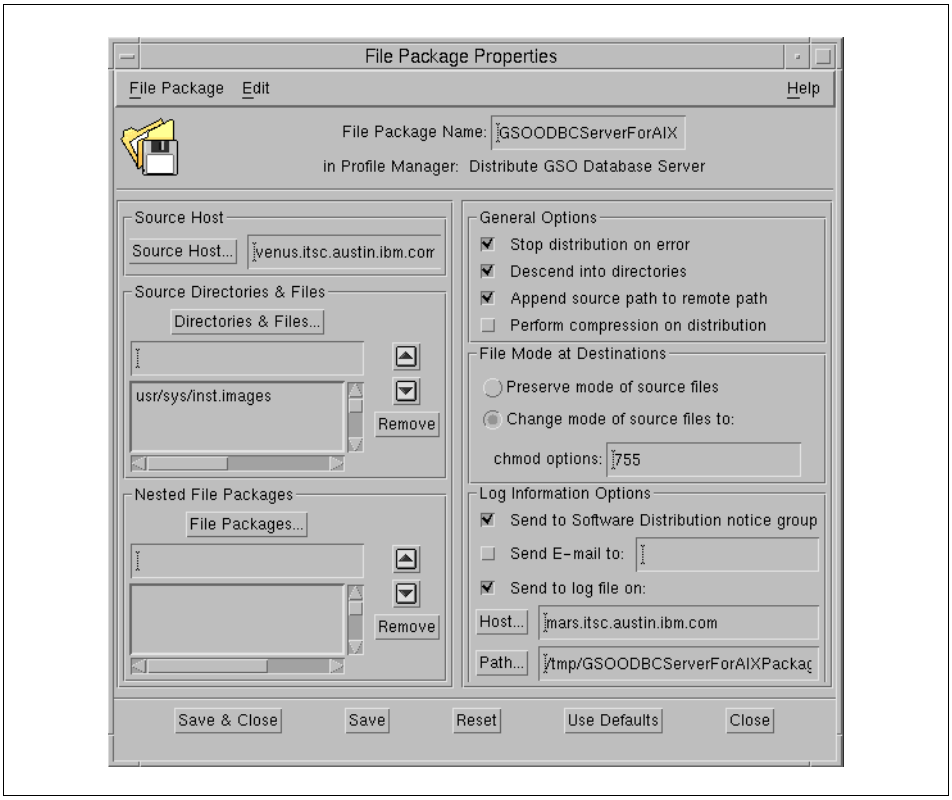


Figure 30. Dialog Displaying the File Package Properties for the GSO Server

The Source Host section displays the host and directory information from where the files are being installed. Do not change this information

because it is automatically chosen from the GSO CD-ROM based on the operating system for which the file package is created.

You should not need to change any options in the General Options sections, unless your environment specifically requires it.

Specify the log information options as required. Ensure that the Host and the Path information under the Log Information Options are set correctly according to your needs. The default path and file name for AIX or Solaris as the operating system is /tmp/<file package name>.log, and in Windows NT, it is c:\Tivoli\db\<hostname>.db\tmp\<file package name>.log.

Click on **Save** if you changed any settings in this dialog to save these changes.

6. Still on the File Package Properties window, select **Distribute...** from the File Package pull-down menu. The Distribute File Package window (Figure 31) appears.

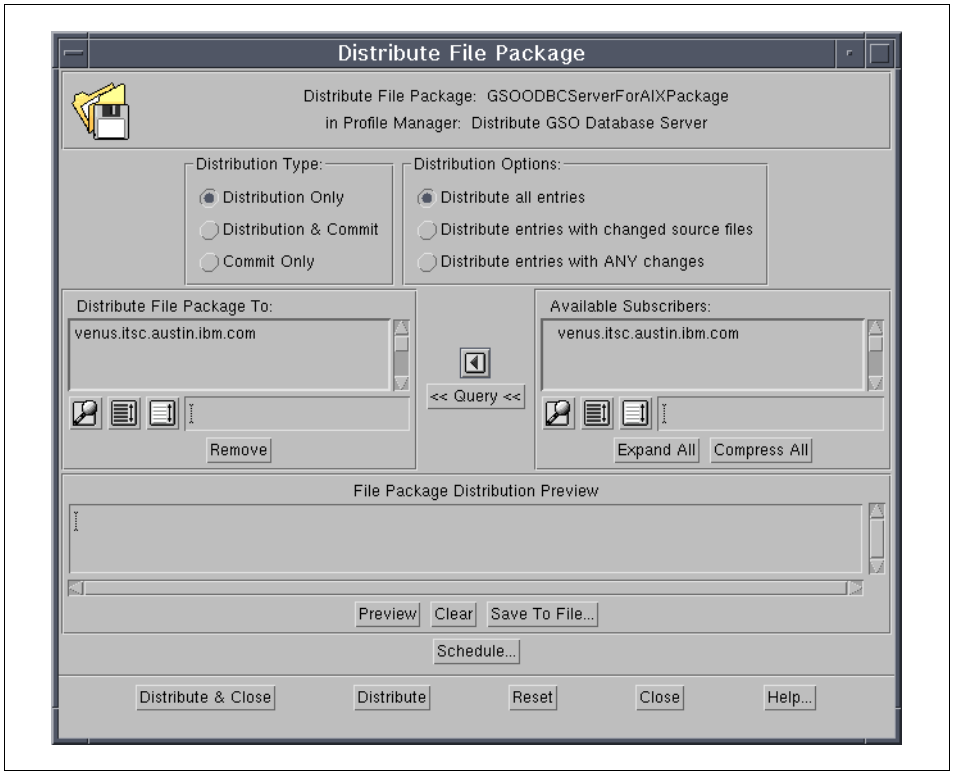


Figure 31. Distribute GSO Database Server File Package Dialog

7. The Available Subscribers list contains the system(s) that have been selected in step 2 above. Move the host(s) to the Distribute File Package To list that are to be installed with the GSO Database Server code. Other fields normally do not need to be changed in this dialog. Click on **Distribute** (or **Distribute & Close**) to start the distribution of the file package to the selected subscribers.

After the file package is distributed, it is important to read through the <file package name>.log file and to check it thoroughly for any errors.

4.4.3 Configuring GSO Servers

The configuration of the GSO server(s) involves the configuration of at least the GSO (master) server, any GSO replica server(s), and any GSO database server(s). A master server is required, all others are optional for a GSO cell to function. Mentioned below are the configuration procedures for all three server types. It is required that the server file packages have been created, distributed and installed successfully according to the previous sections in this chapter before any configuration can take place.

Up until now—hat is, while creating file packages and distributing them to the systems—there was no difference between a master server and a replica server. The difference comes into play only now, when configuring the servers.

Please review section 4.2, “Installation Prerequisites and Considerations” on page 62, before proceeding with the configuration of any of the three server types.

Note on the GSO Cell Name

Configuring a GSO server (a master server, to be more specific) also creates the GSO cell with an associated cell name. Clients can be configured and users can be added to the GSO cell as soon as the GSO master server is configured.

Changing a GSO cell name without loosing any user and client definitions cannot easily be accomplished, and you should therefore very carefully select the GSO cell name so that it does not need to be changed once the GSO cell has been configured and named.

4.4.3.1 GSO Master Server

Configuring a GSO master server involves many steps. Thanks to the Tivoli management environment, it all comes down to two dialog windows that you

need to fill in, and the actual configuration takes place automatically thereafter. Following are the steps necessary to configure the GSO master server.

1. Double-click the **GSO Configuration Tasks** icon from the GSO Plus window. This brings up the Task Library: GSO Configuration Tasks window (Figure 32).

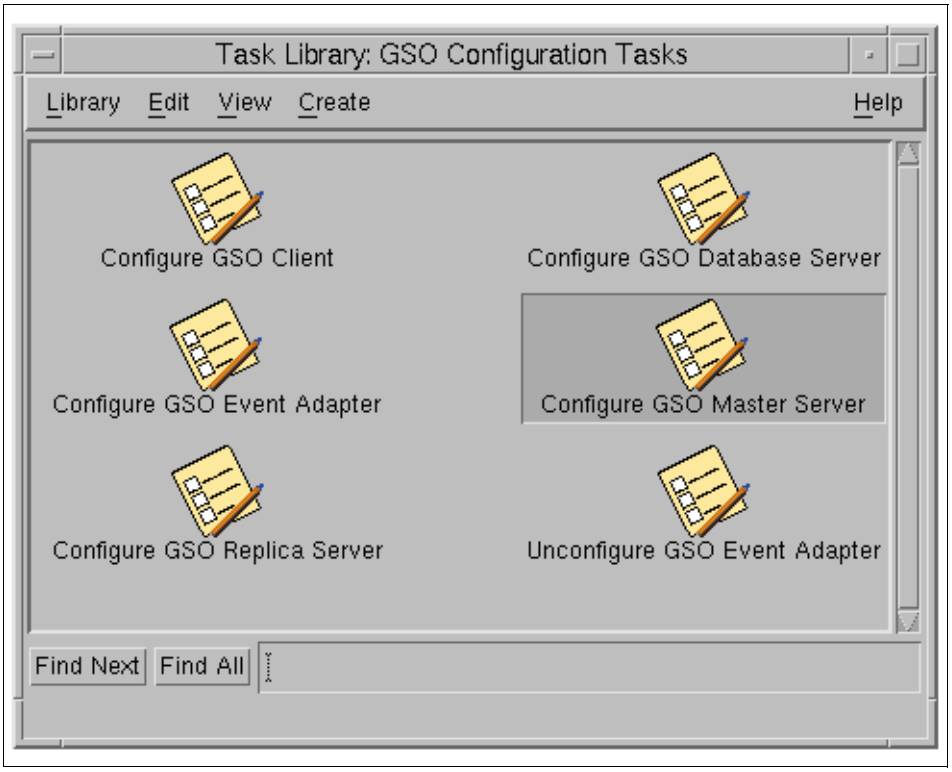


Figure 32. Window Displaying GSO Configuration Tasks

2. Double-click on the **Configure GSO Master Server** icon to launch the Execute Task (Figure 33) window for that particular task.
3. Under the Execution Parameters section, change the Timeout value to 0.
4. Under Output Destination, select **Display on Desktop** so that the configuration process can be monitored on the screen.
5. Under Execution Targets, select the managed node which will be used as the GSO master server by moving the system names between the Available Task Endpoints and the Selected Task Endpoints scrolling lists by highlighting them and using the arrow buttons in between.

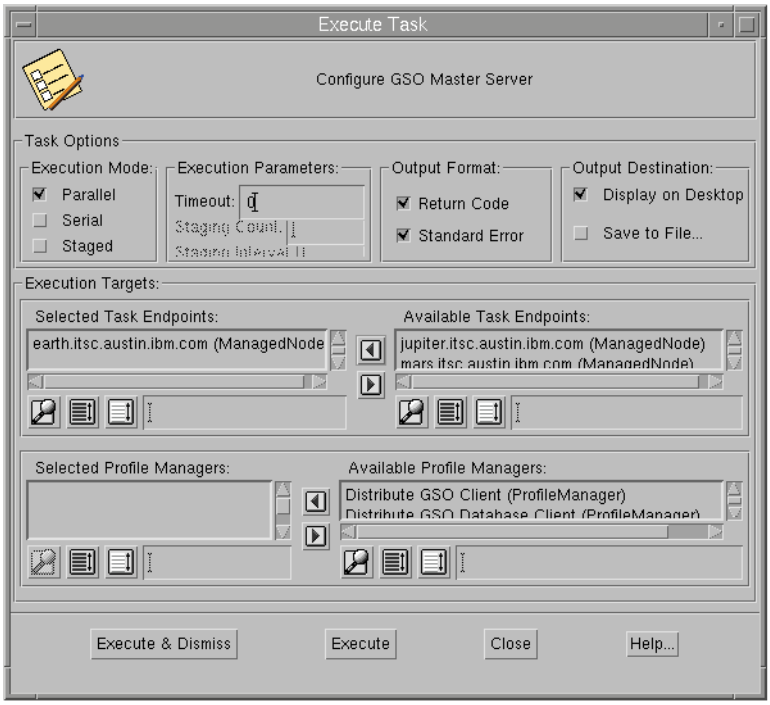


Figure 33. Execute Task Window for Configuration of the GSO Master Server

6. Click on **Execute & Dismiss** and an additional Configure GSO Master Server dialog window pops up, as shown in Figure 34.
7. This is where you type the name of the GSO cell in the New Cell Name field and the password for the cell administrator in the New Cell Password field.

Note on Cell Password

The password that needs to be entered here should be kept confidential. With this password, an administrator (or anybody else) has full authority over the GSO cell.

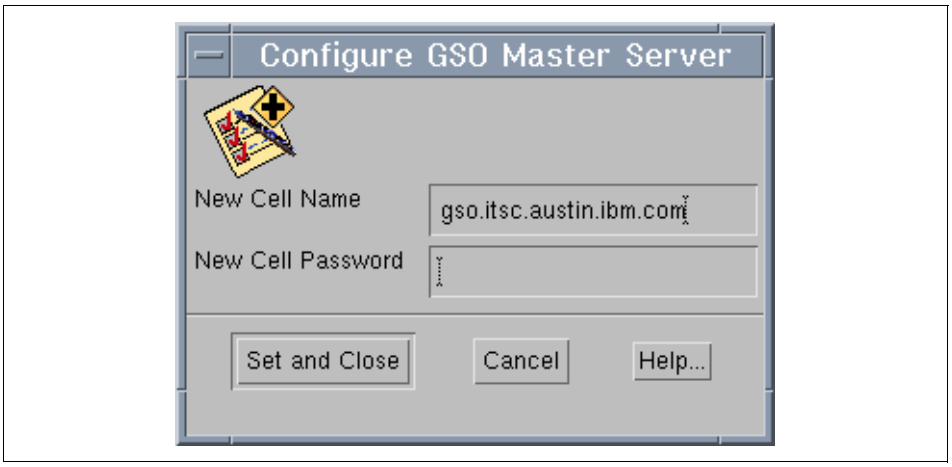


Figure 34. GSO Master Server Configuration Dialog

8. Click on the **Set and Close** button and this commences (and completes) the configuration of the GSO master server.

4.4.3.2 GSO Replica Server

The GSO replica server can only be configured if there is a master server already configured in the GSO cell because this server, in fact, is a read-only copy of the GSO master server. Direct updates and modifications cannot be performed on it directly. Updates or modifications, if any, are automatically replicated by the GSO master server to all replica servers in the GSO cell. There can be any number of GSO replica servers in a GSO cell. The GSO server file package must be distributed to the system on which you are planning to configure a GSO replica server.

Following are the steps to configure a GSO replica server:

1. Double-click the **GSO Configuration Tasks** icon from the GSO Plus window. This displays the Task Library: GSO Configuration Tasks window (Figure 35).

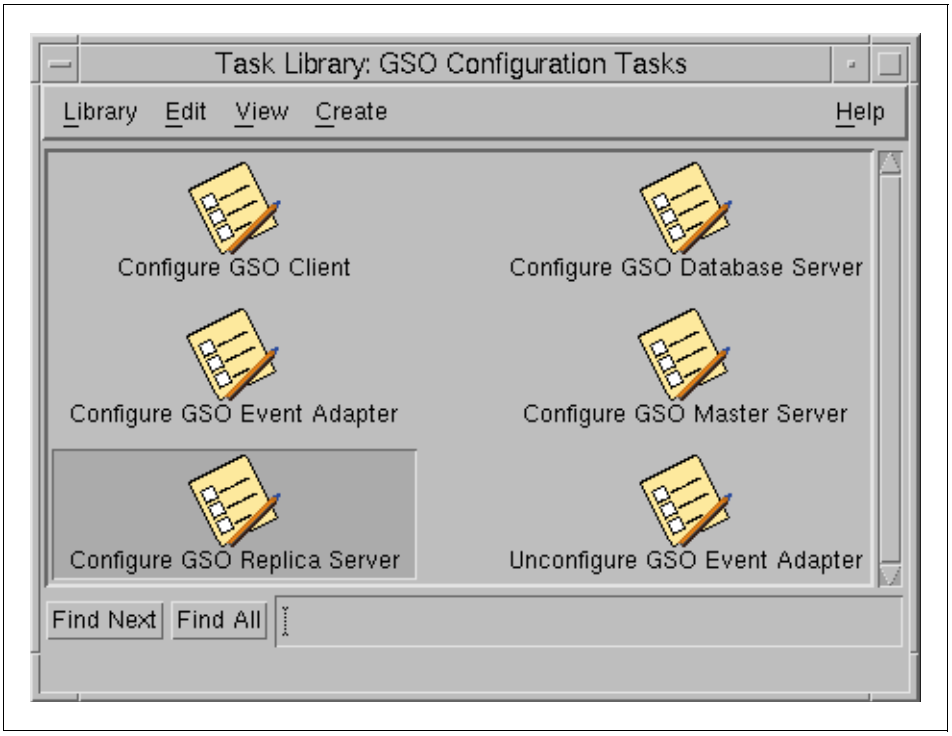


Figure 35. Window Displaying GSO Configuration Tasks

2. Double-click on the **Configure GSO Replica Server** icon to open the Execute Task window (Figure 36).
3. Under the Execution Parameters section, change the Timeout value to 0.
4. Under Output Destination, select Display on Desktop so that the installation process can be monitored on the screen.
5. Under Execution Targets, select the managed node(s) which will be used as the GSO replica server(s) by moving the system names between the Available Task Endpoints and the Selected Task Endpoints scrolling lists by highlighting them and using the arrow buttons between the lists.

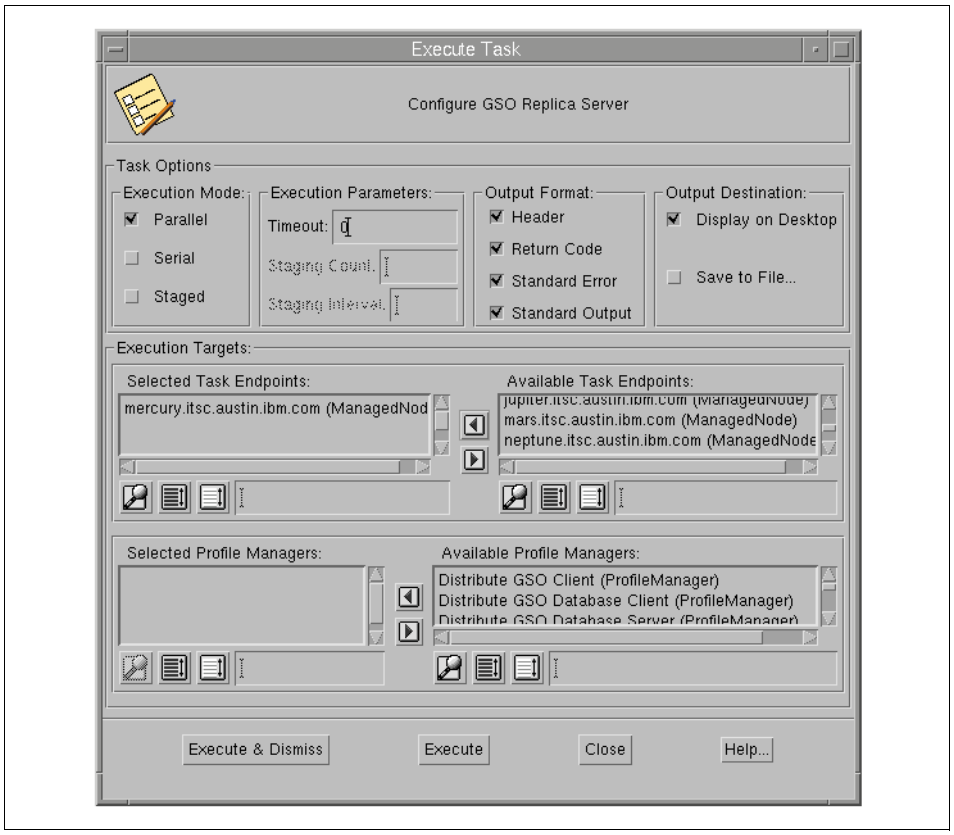


Figure 36. Execute Task Window for Configuration of the GSO Replica Server

6. Click on **Execute & Dismiss** and an additional window, the Configure GSO Replica Server dialog, will pop up (Figure 37).
7. Type the name of the GSO cell in the Cell Name field and the hostname of the GSO master server in the Cell Master Server field.

Figure 37. GSO Replica Server Configuration Dialog

8. Click on **Set and Close**. This completes the configuration of the GSO replica server(s).

4.4.3.3 GSO Database Server

The GSO database server is used by GSO database clients for authentication to the application database. The configuration of this GSO database server is outlined below.

1. Double-click the **GSO Configuration Tasks** icon from the GSO Plus Window. This brings up the Task Library: GSO Configuration Tasks window (Figure 38).

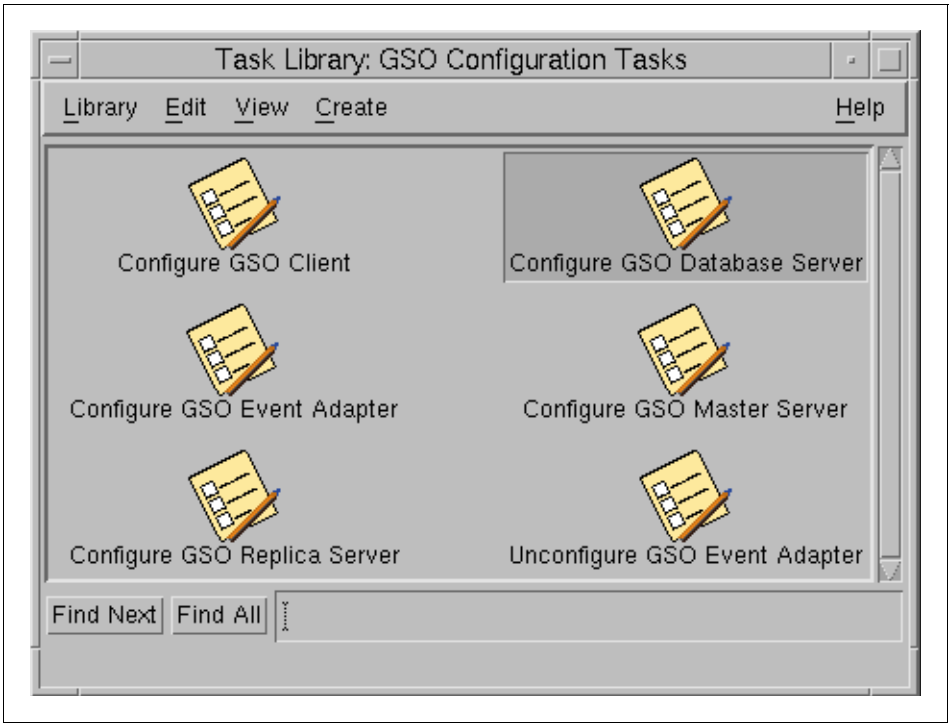


Figure 38. Window Displaying GSO Configuration Tasks

2. Double-click on the **Configure GSO Database Server** icon. This will display the Execute Task window (Figure 39).
3. Under the Execution Parameters section, change the Timeout value to 0.
4. In the Output Destination section, select **Display on Desktop** so that the installation process can be monitored on the screen.
5. Under Execution Targets, select the managed node(s) that will be used as the GSO database server(s) by moving the system names between the Available Task Endpoints and the Selected Task Endpoints scrolling lists by highlighting them and using the arrow buttons between the lists.

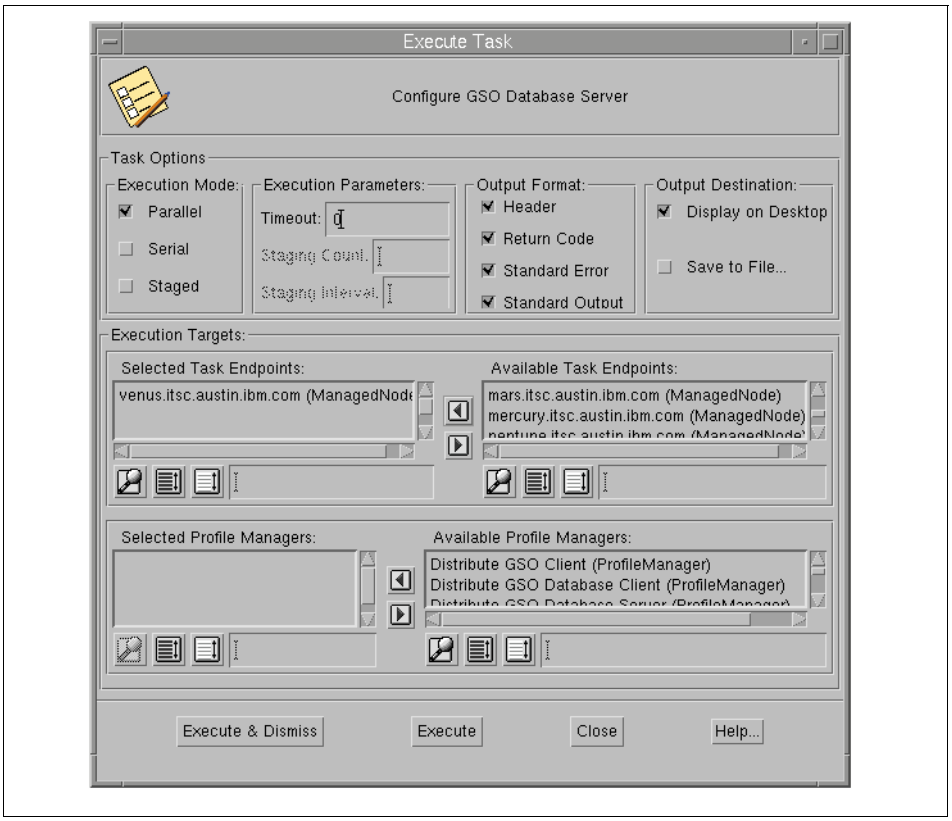


Figure 39. Execute Task Window for Configuration of the GSO Database Server

6. Click on **Execute & Dismiss** and the Configure GSO Database Server dialog will pop up (Figure 40).

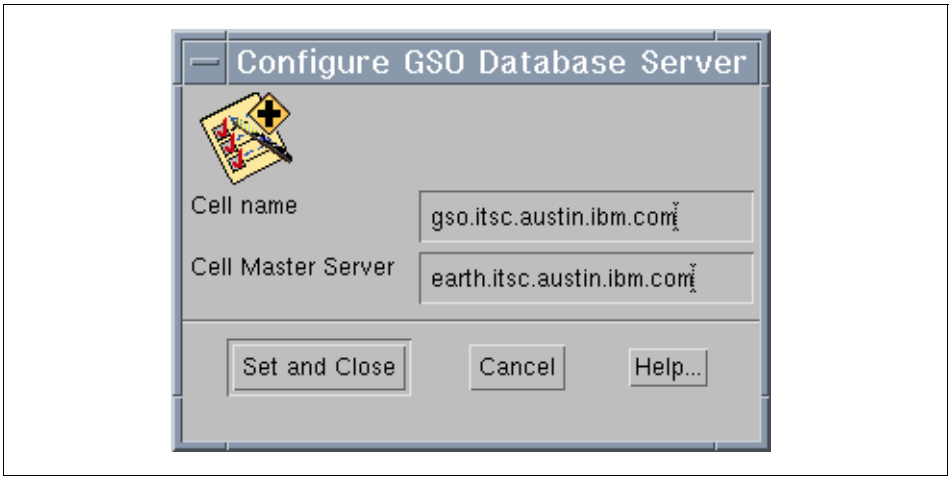


Figure 40. GSO Database Server Configuration Dialog

7. Type the name of the GSO cell in the Cell Name field and the hostname of the GSO master server in the Cell Master Server field.
8. Click on **Set and Close**. This completes the configuration of the GSO database server.

4.5 Summary

Following the steps described in this chapter, you have now installed and configured the Tivoli Management Environment (TME) for use with GSO and at least the GSO master server. The two Tivoli modules, the TME 10 GSO User Administration and the GSO Plus module, have extended the Tivoli framework to allow it to manage GSO resources, events and users. With this in place, you can move on to configure GSO clients, add users to the GSO cell and configure GSO targets.

Although not required, you should consider installation of at least one GSO replica server, unless the installation is for test purposes and only where availability might not be of a primary concern.

Installation of GSO clients is covered in Chapter 5, “Installing GSO Clients” on page 103 and configuring targets is subject of Chapter 6, “Defining Targets” on page 147.

Chapter 5. Installing GSO Clients

The previous chapter explained how to install the different types of GSO servers that may exist in a GSO cell. After having installed and configured the servers, clients need to be set up to complete the installation of a GSO cell. GSO clients, how they work and what functional blocks they contain is described in detail in 2.2, “GSO Clients” on page 15. Below is a short recap of the types of clients in the first section of this chapter.

This chapter explains the installation and basic configuration of GSO clients. Installation and configuration of GSO clients should preferably be done using Tivoli Software Distribution. This is covered in the second section of this chapter. Native installation of clients is covered in 5.3, “Native Client Installation and Configuration” on page 118. This is required, for example, for OS/2 Warp clients. It should be used for Windows NT and Windows 95 clients only when Tivoli Software Distribution is not (and cannot easily be) installed on the workstation. Using the Tivoli Software Distribution is the preferred and recommended way to install GSO clients.

The chapter concludes with a section on Smartcard support and another section that explains how to install and configure biometric authentication support (fingerprint reader).

5.1 Review: GSO Clients and GSO Database Clients

As outlined and explained in 2.2, “GSO Clients” on page 15, GSO clients are machines that provide users physical access to the GSO targets. A target is the final application or subsystem to which GSO signs a user on once he or she is logged on to the workstation and GSO.

GSO 2.0 clients can be either Windows 95, Windows NT, or OS/2 Warp systems. GSO clients cannot provide their services to a user without having access to a GSO server (or GSO replica server). GSO clients run the GSO client code that needs to be installed and configured on the users’ workstations.

There are two types of GSO clients: GSO clients and GSO database clients. The only difference is that a GSO database client runs a database authentication client in addition to the GSO client code that provides for secure authentication to application databases (RDBMSs). GSO database clients need to have at least a GSO database server in the GSO cell to be able to provide this authentication service.

5.2 Installing Clients Using Tivoli Software Distribution

Tivoli Software Distribution supports the concept of file packages. A file package is a descriptive entity that contains all the information necessary to distribute and install a product.

File packages have to be created for both the GSO client and the GSO database client. The GSO database client, if created and distributed to a particular node, would be useless until that client has the GSO client installed, too. This is because the authentication of the user is done using the GSO client software, while the GSO database client initiates and pursues database access. GSO database clients will have to be distributed only to those nodes which require application database access.

As a first step, file packages for the GSO client and the GSO database client have to be created for each platform (refer to 2.4, “Supported Platforms” on page 24). Then these file packages have to be distributed to the client systems based on the function each client will be required to perform and on the operating system.

Mentioned below are the steps that are involved in creation and distribution of the respective file packages.

Table 20. Steps to Installing GSO Clients and GSO Database Clients

Step	Installation Step	Additional Information
1	Create the file package for the GSO client.	For configuration details, see 5.2.1.1, “GSO Client” and GSO product documentation.
2	Create the file package for the database client (optional, only if GSO is used for user authentication to databases).	For configuration details, see 5.2.1.2, “GSO Database Client” and GSO product documentation.
3	Distribute the file package for the GSO client to the managed nodes and PC managed nodes where the GSO clients are installed.	For installation details, see 5.2.2.1, “GSO Clients” and GSO product documentation.
4	Distribute the file package for the GSO database client to the managed nodes and PC managed nodes where the GSO clients are installed (optional).	For installation details, see 5.2.2.2, “GSO Database Clients” and GSO product documentation.

5	Configure the GSO clients on all the managed nodes and PC managed nodes where it has been installed.	For details, see 5.2.3, “Configuring GSO Clients” and GSO configuration product documentation.
6	Configure the GSO database client on all managed nodes and PC managed nodes where it has been installed (optional).	For details, see 5.2.3, “Configuring GSO Clients” and GSO configuration product documentation.

5.2.1 Setting Up Client File Packages

Separate file packages have to be created for each of the two operating systems, Windows NT and Windows 95. Refer to 3.5, “File System Layout and Space Requirements” on page 46, for disk space requirements before starting to set up the file packages.

The creation and distribution of the GSO client file package and the GSO database client file package are similar to the creation of the GSO server file packages (see 4.4.1, “Setting Up Server File Packages” on page 77). It is required that the GSO Plus module be installed as described in 4.3.2, “Installing GSO Plus” on page 72.

When is the GSO CD-ROM Needed?

Creating a file package as described here does not require the actual program files that are to be distributed and installed. Thus, in order to create a GSO client file package does not require the GSO client CD-ROM to be available.

Only when the actual distribution takes place the CD-ROM must be in place according to the definitions within the file package. Alternatively, the files on the CD-ROM could be copied to a hard drive and that directory path could be specified in the file package.

5.2.1.1 GSO Client

Follow the steps below for creation of the GSO client file package:

1. Double-click on the **Tivoli Plus** icon on the Tivoli desktop to open the **Tivoli Plus** window; then double-click on the **GSO Plus** icon to launch the **GSO Plus** window.
2. From the **GSO Plus** window, double-click on the **Set Up GSO Client File Package** icon to open the **Set Up GSO Client File Package** dialog window, as shown in Figure 41.

3. In the File Package Name field, type a descriptive name for the package that you are about to create.
4. In the Source Files Information section, type the host name and directory path where the GSO Client CD-ROM will be available for distribution. Also, select the install language from the selection list.

Figure 41. Setting Up the GSO Client File Package

5. In the Distribution Options section, select the client operating system platform from the selection list, which is either Win 95 or NT.
6. Enter a staging path and select the drive on which you wish to install the client code.

The default distribution staging path is C:\TEMP. This directory is used for temporary storage of the client code during distribution and installation. The directory is created if it does not already exist.

It is necessary to restart Windows NT and Win 95 once the distribution of the file package is complete before it can be used. You can do this manually or by checking the Restart Windows 95/NT after distribution or removal chec-box, in which case the system will be restarted automatically after distribution and installation.

7. Click on **Set and Close** after this dialog is completed. This opens the Set Up GSO Client File Package Output window.
8. The Standard Output section of this window should state successful completion of the creation, and the Standard Error Output section should not indicate any errors. If errors are reported, correct the cause and, if required, redo the file package creation.
9. If you want to store the contents of the dialog, click on the **Save to File...** button. If you do not wish to save the dialog, click on the **Close** button.

Timeout Errors?

Creating a client file package is a Tivoli job with an assigned timeout value. If the system being used is heavily loaded or slow for any other reason, creation of a file package can abort with a timeout error indication. This timeout value can be modified to prevent such errors from happening. Right-click on the **Set Up GSO Client File Package** icon in the GSO Plus collection window and select **Modify job...** from the pop-up menu. In the upcoming Edit Job dialog window, locate the Timeout value and increase it according to your needs (or specify 0 for indefinite).

This completes the creation of a file package for a GSO client, specific to the operating system chosen.

Two separate packages have actually been created, and you will see two new icons in the Distribute GSO Client collection window (right-click on the icon, then select **Open...** from the pop-up menu to see the contents of this collection). Every GSO client file package that you create causes another package with the name suffix “Nested” to be created automatically. This is for the purpose of specific distribution implementation, and you should not use the nested package for any operations; it is referenced by the primary file package. Only when you delete a client file package, should the accompanying nested file package be deleted.

5.2.1.2 GSO Database Client

Follow the steps mentioned below for creation of the GSO database client file package:

1. Double-click on the **Tivoli Plus** icon on the Tivoli desktop to open the **Tivoli Plus** window, then double-click on the **GSO Plus** icon to launch the **GSO Plus** window.
2. On the window that appears, double-click on the **Set Up GSO Database Client File Package** icon to launch the Set Up GSO Database Client File Package dialog window (Figure 42, taken from a Windows NT server).

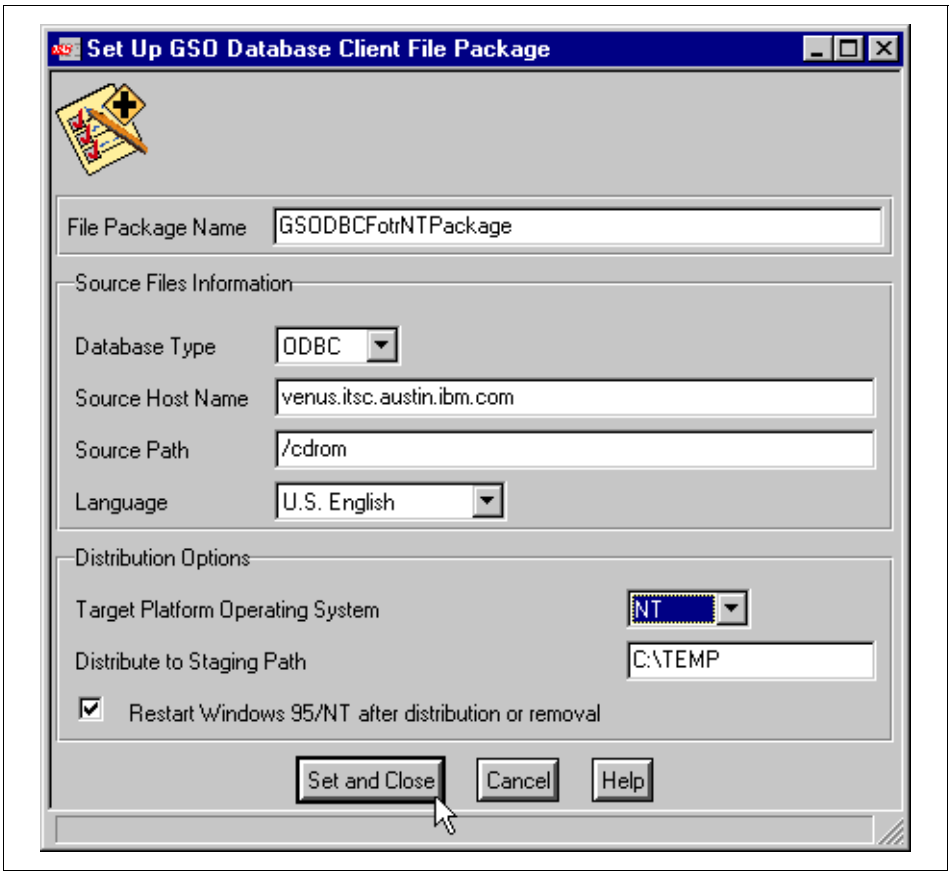


Figure 42. Setting Up the GSO Database Client Package

3. In the File Package Name field, type a descriptive name for the package that you are about to create.

4. In the Source Files Information section, select the database access type (ODBC, Oracle, or Sybase), the source host name and the directory path where the GSO client CD-ROM will be available for distribution. Then, select the client code language from the selection list.
5. In the Distribution Options section, select the target platform operating system from the list (Win 95 or NT) and specify the distribution staging directory on the client(s) for temporary storage of files during distribution and installation.

   Checking the **Restart Windows 95/NT after distribution or removal** check box will cause the client to be restarted automatically after the client software was distributed and installed. This is necessary, but can be done manually as well.
6. Click on **Set and Close** after you have completed this dialog. This will open the **Set Up GSO Client File Package Output** window.

   The Standard Output section of this window should state successful completion of the file package creation. The Standard Error Output section should not indicate any errors.
7. If you want to store the contents of the dialog, click on the **Save to File...** button. If you do not wish to save the dialog, click on the **Close** button.

Note: If you get timeout errors, read the note on page 107.

This completes the creation of a file package for the GSO database client for the operating system chosen. As with GSO client file packages (see last section), two file packages (and two icons) have been created: one with the name that you have chosen and the other with the same name and with “Nested” appended to it.

5.2.2 Distribution of GSO Client File Packages

After the GSO client file packages have been created as described in the last sections, they need to be distributed to the client machines in order to get installed. Remember that a file package only is a descriptive set of information about the package; it does not contain the actual installable code. For this reason, the media that contains the code (normally the GSO client CD-ROM, unless its contents has been copied onto a disk) must be available during distribution.

Another prerequisite, of course, is that the client systems must have the adequate Tivoli software components installed, configured and running (see also 2.4, “Supported Platforms” on page 24, and the product documentation for a list of prerequisites).

5.2.2.1 GSO Clients

After you have created the GSO client file packages, the Distribute GSO Client collection (contained in the GSO Plus collection) lists them as icons. When you open it (right-click on the **Distribute GSO Client** icon, then select **Open...** from the pop-up menu), it might look like the one shown in the following Figure 43.

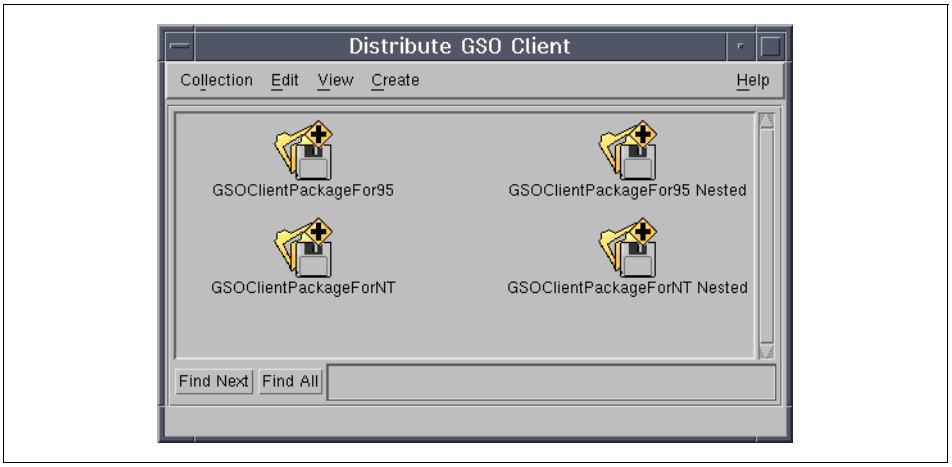


Figure 43. GSO Client File Packages

As can be seen in Figure 43, two file packages have been created rather than just one for every create-file-package operation. Although required for the distribution, we do not need to bother about the ones with the “Nested” name extensions. Follow the steps below to distribute a GSO client file package:

1. A first step now is to define a set of client machines that potentially can be GSO clients. Right-click on an icon and select the **Subscribers...** option from the pop-up menu. Note that because of the way Tivoli/Plus modules work, you can choose any icon in this window (Figure 43) as the subscribers list applies equally to all file packages. This opens the Subscribers window shown in Figure 44.

   Notice that the Subscribers dialog window may contain a long list of potential subscribers in the right-hand list that includes not only available managed nodes but also other potential subscribers. You should select the managed nodes that are to be installed from that list and move them to the left-hand list under Current Subscribers. (If the lists are too long to manually search for a particular host, you may use the search tools provided in the lower part of that dialog or use the Query tool. See the Tivoli documentation for more information on how to use them.) Click on

Set Subscriptions & Close to save the selections and dismiss the window.

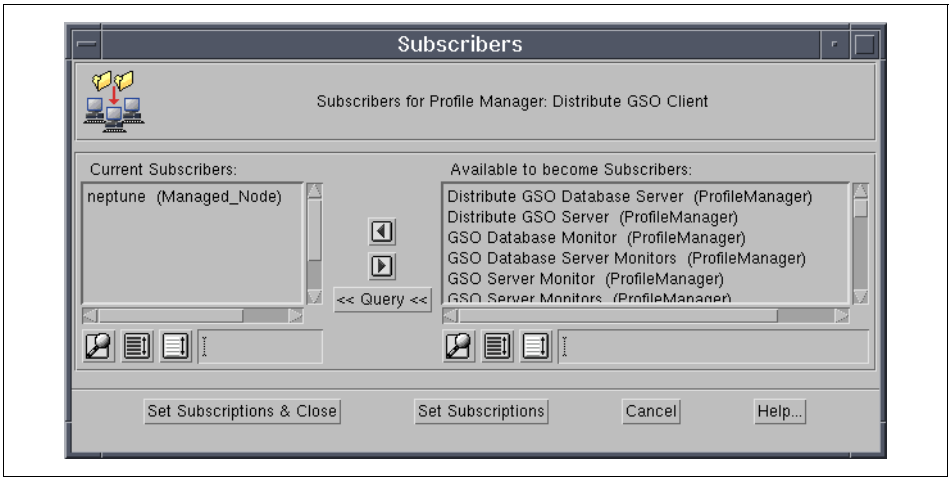


Figure 44. Subscribers for the GSO Client File Package

2. As a next step, you should open the File Package Properties dialog by right-clicking on the icon representing the file package that is supposed to be distributed (Figure 43) and then selecting the **Open...** function from the pop-up menu. An example File Package Properties dialog is shown in Figure 45. Some of the information shown in this dialog, such as source host, directory and nested file package information, was automatically defined when the file package was created. You would not normally be required to alter any properties in most portions of that dialog, but you might want to change the logging options contained in the Log Information Option fields to suite your needs.

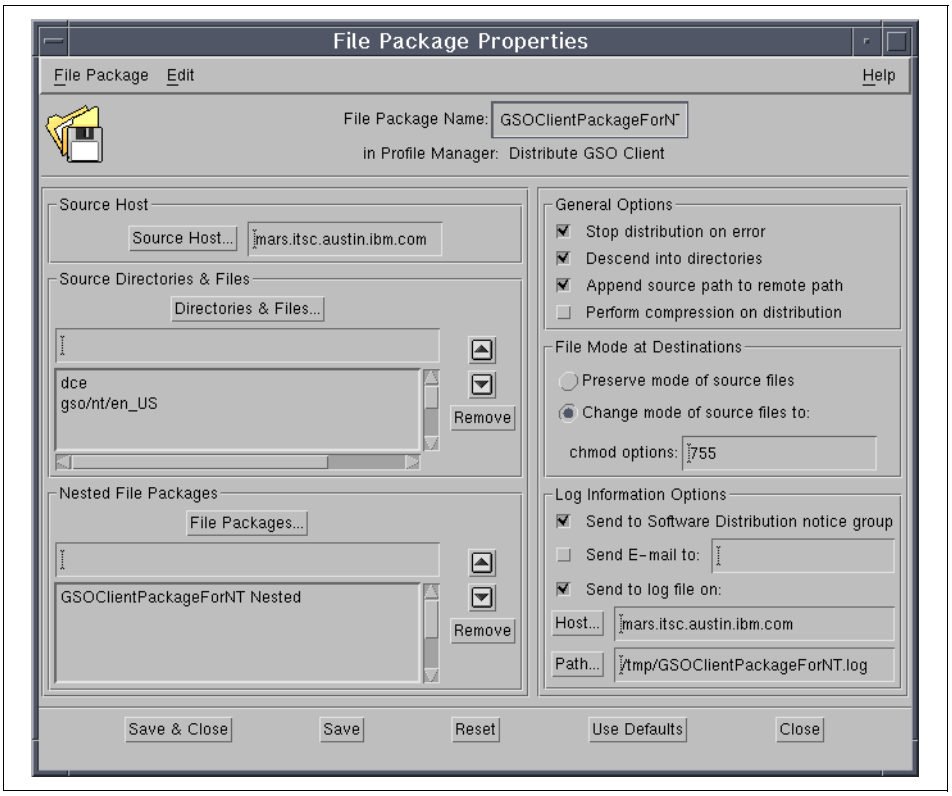


Figure 45. File Package Properties Dialog

Make sure you save any changes made on this dialog by clicking on **Save** (or **Save & Close**).

3. After the file package properties have been reviewed (or changed and saved), you should check some prerequisites before you distribute the file package, such as:
   - The TMR server and the client(s) must be operational, including the Tivoli software components, and they must be able to communicate over the network.
   - There must be enough free disk space available on the client (see 2.5, “Hardware Requirements” on page 26).
   - The GSO client CD-ROM (or the equivalent file path if the contents of the CD-ROM was copied onto a hard drive) must be accessible through the path and host that are specified in the file package.

4. Select the **Distribute...** option from the File Package pull-down menu in the File Package Properties window (Figure 45). This launches the Distribute File Package window shown in Figure 46.

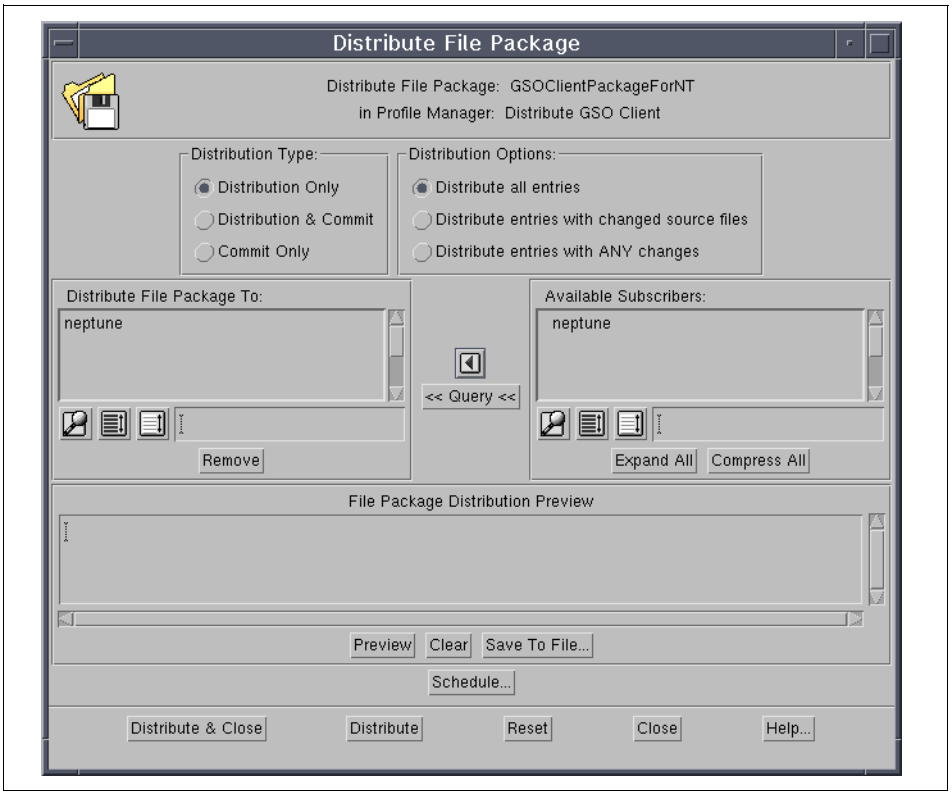


Figure 46. Distribute File Package Dialog

In this dialog, move the destination hosts from the Available Subscribers list to the Distribute File Package To list on the left-hand side. Other options need not normally be changed. Notice, however, that there is a Preview options that allows you to review the files that are to be copied to the client.

5. After the destination system(s) are selected, click on **Distribute** (or **Distribute & Close**) to start the distribution process. Alternatively, you could choose the distribution to take place automatically at a given time in the future by clicking on **Schedule...** and entering the necessary information on the Add Scheduled Job window that appears.

This last step starts the actual distribution of the GSO client file package.

Note on Distribute

The file packages icons in the various collection windows, such as the one shown in Figure 43, also have a **Distribute** option in their pop-up menus when you right-click on them. Be very careful using them as they immediately start a distribution process for all defined file packages, not only for the one from which you started the operation.

The distribution process runs in the background, and you do not get an immediate feedback other than a message on the Tivoli desktop indicating the start and finish of the distribution operation. The result of the distribution, however, can be seen in the log file as specified in the File Package Properties window (Figure 45 on page 112). After a successful distribution, the log file might look like the following example:

```
# cat /tmp/GSOClientPackageForNT.log
File Package:  "GSOClientPackageForNT"
Operation:     install  (m=5)
Finished:      Tue Oct  6 19:31:04 1998
--------
Source messages:
<none>
--------
neptune:SUCCESS
temp script: nt_before: c:\var\spool\Tivoli\neptune.dbmp\nt_b2.bat
temp script: nt_after: c:\var\spool\Tivoli\neptune.dbmp\nt_a3.bat
temp script: nt_removal: c:\var\spool\Tivoli\neptune.dbmp\nt_r4.bat
starting script: c:\var\spool\Tivoli\neptune.dbmp\nt_b2.bat
script complete: exit code=0
C:/TEMP/gso/gso/nt: creating path
starting script: c:\var\spool\Tivoli\neptune.dbmp\nt_a3.bat
script complete: exit code=0
==============
#
```

The log file should be looked over in any case to determine whether or not the distribution was successful. If the distribution failed, the log file contains additional information about any errors that occurred. Common errors might be that there is not enough free space in the temporary staging directory or that a file could not be found because the GSO client CD-ROM was not inserted in the drive at the time of the distribution.

Remember that the client system(s) need(s) to be rebooted after this distribution. If automatic restart was selected when the file package was created (see previous sections), this will automatically be initiated after the distribution.

Alternative Ways for File Package Distribution

The Tivoli Framework offers many flexible ways to organize your jobs, administrators and the infrastructure to be managed. The step described above is just one way to distribute file packages. Another way could be to open the Distribute GSO Client profile manager from the GSO Plus top-level policy region. It contains all the created GSO client file packages that can be distributed using the profile managers distribute functions. Profile managers also offer other functions, such as calculating the package file size. You might also have reorganized policy regions and created your own profile managers for distribution. It is beyond the scope of this redbook to describe the many features Tivoli offers. See the Tivoli documentation for additional information on how policy regions and profile managers can help organize your work and how they can be used for file package distribution.

5.2.2.2 GSO Database Clients

Distribution of GSO database clients can be done exactly the same way as when distributing GSO clients (see section 5.2.2.1, “GSO Clients” on page 110). A GSO database file package may only be distributed to systems that have the GSO client packages installed. Before distribution, check the system requirements in 2.5, “Hardware Requirements” on page 26 and the GSO documentation (including the Readme files supplied with the product CD-ROM). Make sure that you distribute the correct file package for the GSO database client appropriate to the database being used.

5.2.3 Configuring GSO Clients

After distribution of the file packages, GSO clients need to be configured before they can be used. Configuration of a GSO client includes:

- Providing connection information for GSO servers
- Enabling or disabling integrated login
- Configuring Litronic Smartcard support, if necessary

Note that this configuration does not apply to GSO database clients.

These configuration tasks could be done on every workstation using the `cfgclient` command, or, more conveniently, from a central management point by using Tivoli tasks.

To configure a GSO client (or a group of GSO clients), follow the steps below:

1. Open the GSO Plus collection window. To do this, double-click on the **TivoliPlus** icon on the TME desktop to open the TivoliPlus window; then double-click on the **GSO Plus** icon in this window.
2. Double-click on the **GSO Configuration Tasks** icon to open that task library window.
3. Double-click on the **Configure GSO Client** icon. This launches the Execute Task dialog window shown in Figure 47.

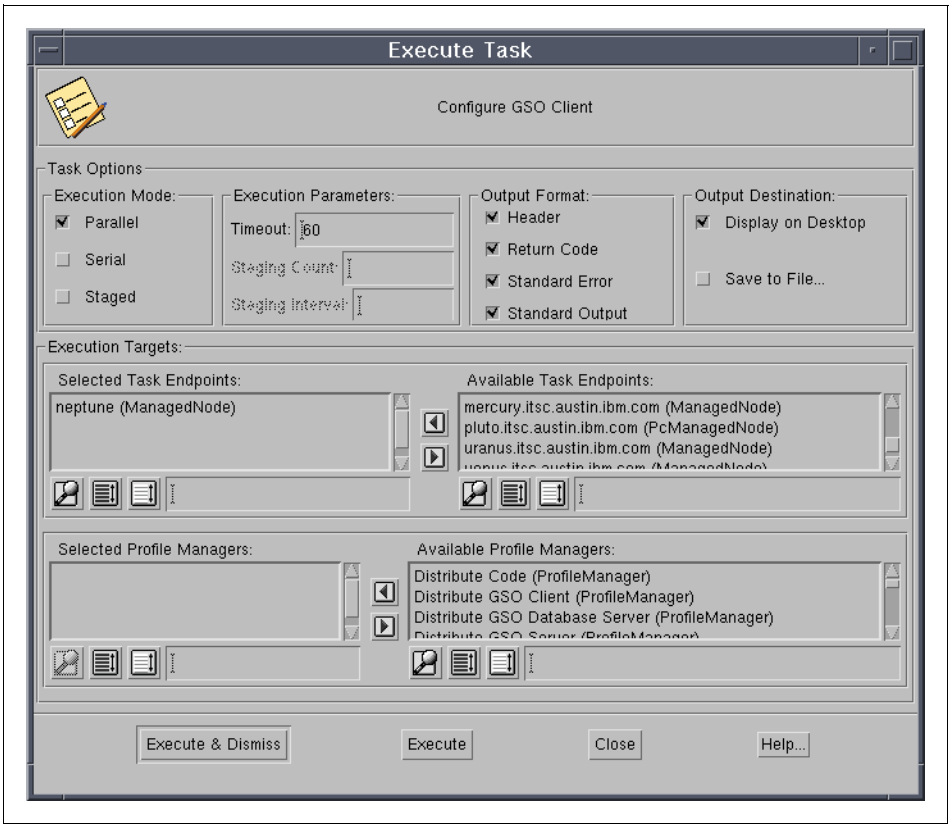


Figure 47. Execute Task Dialog for GSO Client Configuration

4. You need to specify the client system(s) that is (are) to be configured by moving the hostname(s) from the Available Task Endpoints list to the Selected Task Endpoints list. This can be a single or multiple GSO client(s). If you suspect slow operations, for example due to slow network links, you may increase the Timeout value or set it to 0 (zero) to disable the Timeout check. You may also specify that the output be displayed on the desktop and/or be saved in a file. Once you have completed this

dialog, click on **Execute** (or **Execute & Dismiss**). The Configure GSO Client dialog window appears (Figure 48).

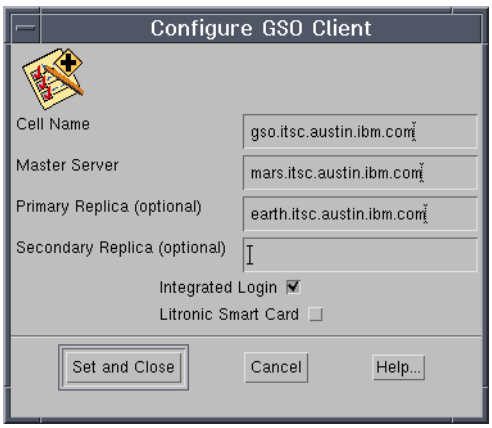


Figure 48. GSO Client Configuration Dialog

5. Fill in the information as required and select whether or not the client(s) will use integrated login and Litronic Smartcard support. The cell name is the name that was assigned to the GSO cell when the GSO master server was configured (see 4.4.3.1, “GSO Master Server” on page 92). The name of the GSO master server and (optionally) two more GSO servers can be specified. Additional GSO servers increase the availability of the login service.
6. After completing the Configure GSO Client dialog, click on **Set and Close** to start the configuration task.

This initiates GSO client configuration. If you selected that output be displayed on the desktop, a window opens that contains the messages from this task. Figure 49 shows an example output screen as a result of a successful client configuration.

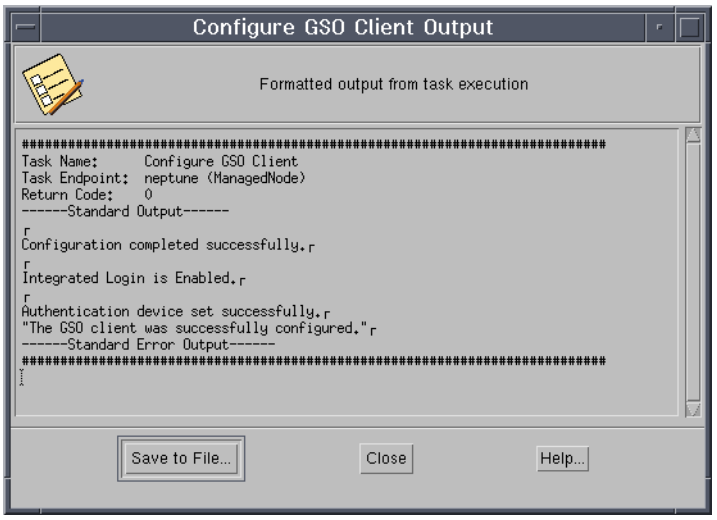


Figure 49. Formatted Output from Configure GSO Client

You should check that there is no indication of an error and that the Standard Error Output section is empty (or does not indicate a serious error).

This concludes installation and configuration of GSO clients. The next sections describe native installation—that is, installation of GSO clients without using the Tivoli Framework.

5.3 Native Client Installation and Configuration

As pointed out earlier, using Tivoli Software Distribution is the preferred and recommended method for installing GSO clients (except for OS/2 Warp, where there is only the native method as described here). Native client installation might be an option in a test environment where full Tivoli support is not deployed.

5.3.1 Windows NT and Windows 95 Clients

Following is a step-by-step description of the native client installation for Windows NT and Windows 95. The IBM GSO client component is made up of a DCE component and the GSO client component itself.

First, the DCE component of the IBM GSO client has to be installed:

1. Insert the GSO Windows NT/95 client CD-ROM in the CD-ROM drive of this client machine.
2. Click on **Start** -> **Run** and enter

   `x:\dce\setup`

   as the program to run, where `x:` is the drive letter assigned to your CD-ROM drive. Then, click on **OK** to start the installation.
3. A Choose Setup Language window will now appear. Click on the language that you prefer to have the installation done in and click on **Next**.
4. You will now get a welcome window for the installation of Intraverse NetSEAT DCE client Version 2.1. Click on **Next** to continue.
5. Specify a directory where you would like to install the DCE client software. It is recommended to install the software in the default directory, which is c:\netseat. Then click on **Next**.
6. You will now get the Start Copying Files window that displays the settings that you have chosen so far. Use the **Back** button if you need to modify these settings. If they are correct, click on **Next** to start the installation process.

   The product files are then copied to the hard disk. Once the copying of files is complete, you will get the setup complete pop-up.
7. Windows needs to be restarted after installation. Click on **Yes, I want to restart my computer** if you want the computer to be restarted immediately, or choose **No, I will restart my computer later** otherwise. It is required to reboot your computer before you proceed further in your installation process. After clicking on either option, click on the **Finish** button to finish the installation.

After the system has been restarted, the installation of the GSO client component has to be done:

1. Ensure that the GSO client CD-ROM is available in the CD-ROM drive of the system.
2. Click on **Start** -> **Run** and enter

   `x:\gso\nt\setup`

   on Windows NT or

   `x:\gso\95\setup`

   on Windows 95 as the program to run, where `x:` is the drive letter assigned to your CD-ROM drive and click on **OK** to start the installation process.

3. Choose the desired language from the window that appears and then click on **Next**.
4. Click on **Next** on the welcome window for the GSO client software.
5. Specify a directory where you want to install the GSO client software. It is recommended to install the software in the default directory, which is c:\ibmgso. Then, click on **Next**.
6. You will now get the Start Copying Files window that displays the settings that you have chosen so far. Use the **Back** button if you need to modify these settings. If they are correct, click on **Next** to start the installation process.

   The product files are then copied to the hard drive.
7. After copying of files is complete, you will be given the options to read the README file and to add GSO to the startup folder. It is always a good idea to review the README file, and it is also recommended to add GSO to the startup folder to have it started automatically rather than manually. Click on **Next** to continue.
8. As a last step, the client computer needs to be restarted, either automatically or manually. Choose the appropriate option and click on **Finish** to complete the installation. Make sure the machine is restarted before continuing with the configuration.

After the NetSEAT and GSO software has been installed, the client needs to be configured and added to the GSO Cell. This is done using the `cfgclient` command on the client. This command can be found in the <ibmgso>\bin directory on the client (<ibmgso> is the GSO install path as specified during installation; by default it is c:\ibmgso).

The `cfgclient` command has a number of options. These can be displayed by entering `cfgclient` at the command line, as shown in the following panel (Figure 50).

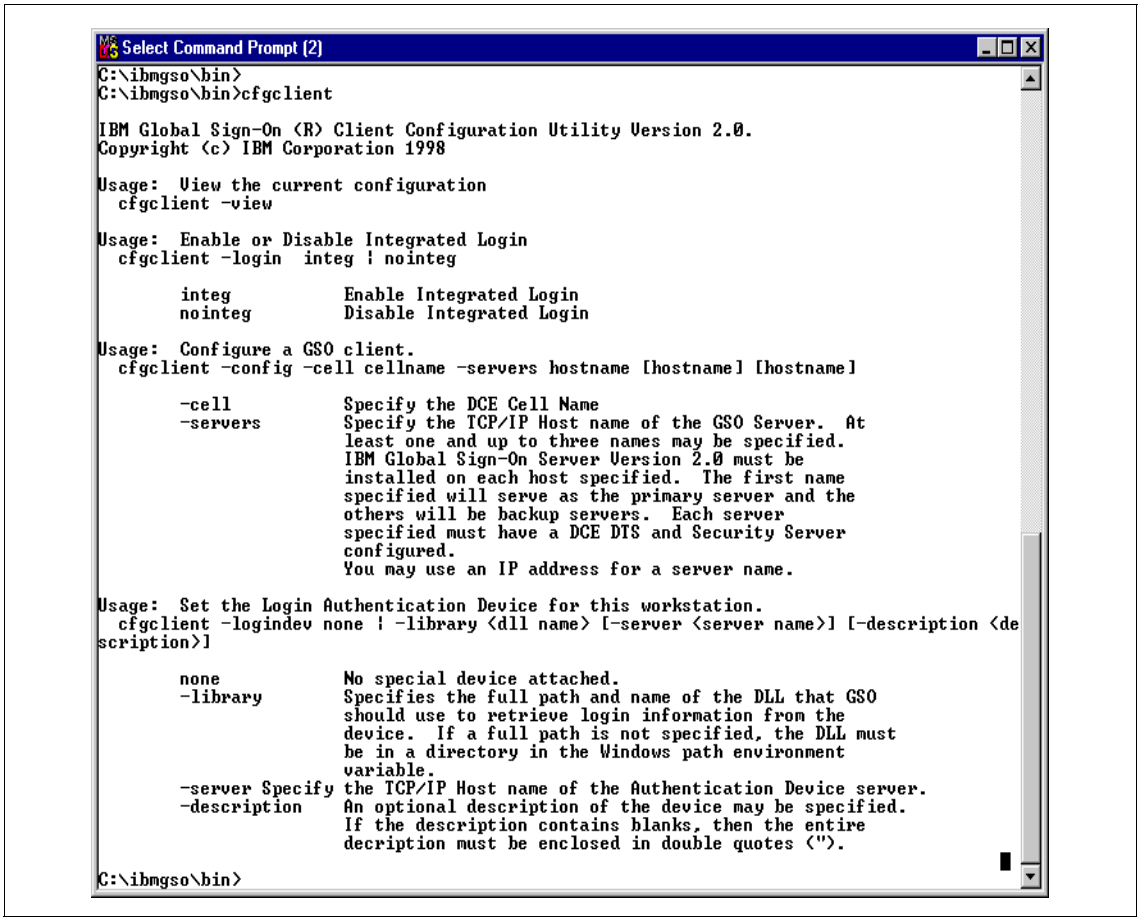


Figure 50. GSO cfgclient Command Line Options

The options to configure the logon device are used for Smartcards and biometrics. These are covered later in this chapter in 5.4, “Adding Smartcard Support” on page 130, and in 5.5, “Adding Biometric Support” on page 138, respectively. You may check the current setting of integrated login and configure it as required in your environment.

At this point, we need to configure the GSO client into the GSO cell. To do this, enter the following command:

```
cfgclient -config -cell <GSO cell name> -servers <GSO server(s)>
```

where:

`GSO cell name` is the name of the GSO cell as specified when the GSO master server was configured (see 4.4.3.1, “GSO Master Server” on page 92).

`GSO server(s)` is a list of one to three hostnames of GSO servers that the client can connect to.

After this configuration command has completed, the GSO client is configured, and you will be able to use the GSO Launcher and GSO Administration GUIs to log on to GSO.

5.3.2 OS/2 Warp Clients

The IBM Global Sign-On for Multiplatforms, Version 2.0 GSO Plus module for Tivoli does not include the file package set up and distribution or client configuration task for OS/2 Warp.

The following steps detail how to install and configure the OS/2 Warp clients. Note that once the OS/2 Warp clients have been installed and configured into the GSO cell, the GSO Launcher and GSO Administration GUIs on the workstation function in the same way as they do for Windows NT and Windows 95 clients. GSO users and targets are administered from Tivoli exactly as they are for the Windows client platforms.

The first step to installing GSO on an OS/2 Warp client machine is to install the DCE client. Included on the GSO client CD-ROM for OS/2 Warp is the IBM DCE client. The description that follows details how to install and configure the DCE client on OS/2 Warp for GSO. This does not preclude configuring other DCE client components later on, such as a DFS client, if these are needed for other applications. It also does not preclude installing GSO on an OS/2 Warp workstation that is already running DCE. Providing the DCE client has been configured into the same DCE cell that will be running GSO, you may omit the installation of DCE and simply install GSO.

The DASCOM Intraverse NetSEAT DCE client provided with GSO for Windows NT and Windows 95 clients is not available for OS/2 Warp. However, the IBM DCE client does include a slim client option as well as the full DCE client. If you are installing DCE purely for GSO, the slim client is recommended because it requires less disk space and memory.

Before you can install DCE, you need to disable the FixPak check on Warp 4.0 (due to a known program error). The operating system prerequisites are OS/2 Warp Version 3.0 with FixPak 21 or higher or OS/2 Warp Version 4.0. To disable the FixPak check on Warp 4.0; enter `SET DSSFPCHK=1` in the command

prompt window where you will be running the installation. Then, follow these steps:

1. Insert the OS/2 Warp GSO client CD-ROM into the CD-ROM drive. Change to the drive root directory and type `install` on the command line.
2. After the initial welcome screen, the following pop-up window appears, asking for the components to configure (Figure 51). Check the boxes to install **DCE Base Services** and (optionally) **DCE Books**. If you wish to install on a drive other than C:, use the **Target Drive** option on the right-hand side of the screen to specify the drive you require. If you wish to check disk space or change the default path names that will be used, click the **DASD..** button at the base of the screen. Enter the values you require on the panel that is displayed and click **OK** to begin the installation or **Cancel** to return to the main screen.

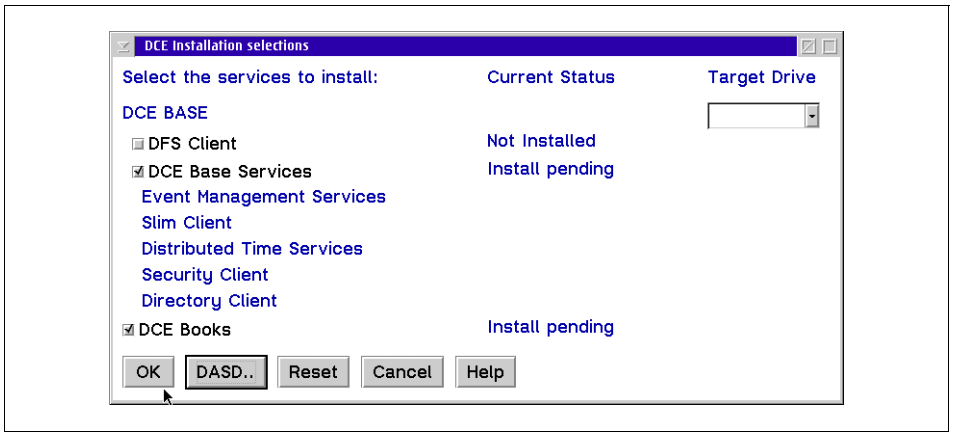


Figure 51. OS/2 Warp Install Options

3. When the installation has completed, a panel will be displayed listing the components that have been installed. You should click **OK** to end the installation. You must then reboot the workstation in order to the changes to take place.
4. Before continuing with the DCE configuration, TCP/IP must be installed and configured on that client machine. One method to view the TCP/IP configuration is to double-click on the **OS/2 System** icon on the desktop. Then double-click **System Setup** and then double-click **TCP/IP Configuration (LAN)**. This opens the TCP/IP Configuration window (Figure 52).

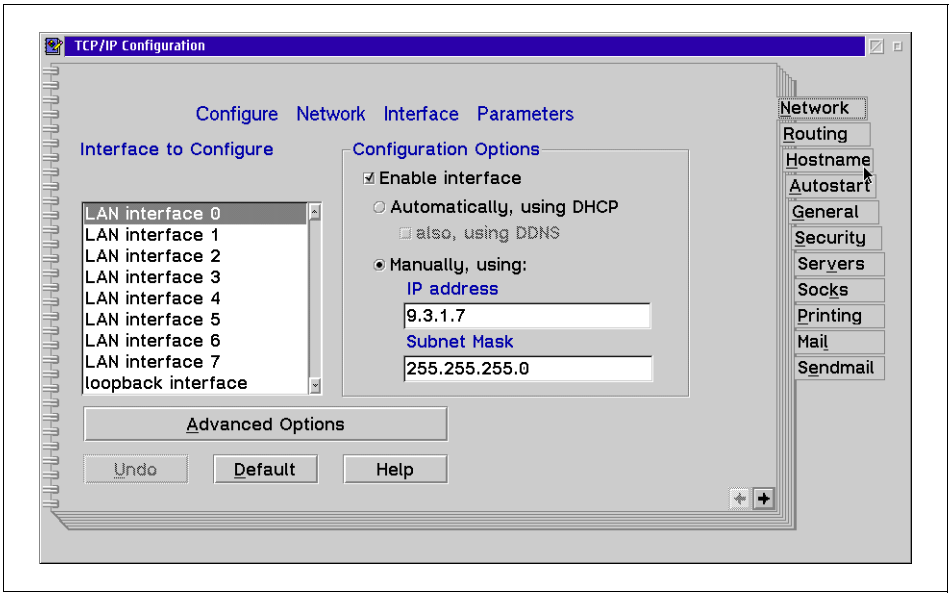


Figure 52. OS/2 Warp TCP/IP Configuration

5. The Configure Network Interface Parameters panel (shown above) will be displayed as the first screen. Make sure that the **Automatically, using DHCP** option under **Enable interface** has NOT been checked. DCE does not support this option at the time of writing. It may be worth checking the latest product documentation to see if the support has been added in your release or version. Meanwhile, you should check the **Manually, using** option and enter the IP-address and Subnet Mask values.
6. Click the **Hostname** tab on the right-hand side of the screen. The Configure LAN Name Resolution Services screen will be displayed. Ensure that the Hostname and Domain fields have been completed with the values for this workstation. You also need to ensure that a Nameserver Address has been entered if you use a name server. As a special note, the host name for the workstation must be entered here as well, even if it is set in your config.sys file.
7. Click the **General** tab on the right-hand side of the screen to display the Configure General Parameters screen. Ensure the correct **Timezone** has been selected. This parameter can be set through the config.sys file; however, since TCP/IP configuration updates the config.sys, it makes sense to ensure this field is set correctly.
8. If any changes are made to the TCP/IP configuration, you will need to reboot the workstation at this time for them to take effect.

9. NetBIOS is required in addition to TCP/IP. To check if it is installed and configured, from the desktop click **OS/2 System** -> **System Setup** -> **MPTS Adapters and Protocol Services** -> **Configure.** Check that **NetBios Socket access** shows as configured. If it does not, you will need to install and configure it.

10. In addition to checking the TCP/IP and NetBIOS configurations, it is vital to ensure the workstation is running the correct date and time. If the difference in time between the workstation and the DCE server is greater than five minutes, the DCE configuration will fail with a `clock skew too great` error.

    In an IBM LAN Server environment, for example, the clocks on the client machines are synchronized with the domain controller(s) during logon. This function may need to be disabled to prevent the client machine from becoming out of synchronization with the DCE server unless there is a reliable time synchronization mechanism between the LAN server domain and the DCE server. To do this, you need to edit the ibmlan.ini file on the client machine and change byte 36 of the WRKHEURISTICS field to zero.

11. Once the TCP/IP and NetBIOS environment has been set, you can configure DCE. Double-click on the **Configure DCE Services** icon under the **DCE Services** icon on the desktop (Figure 53).

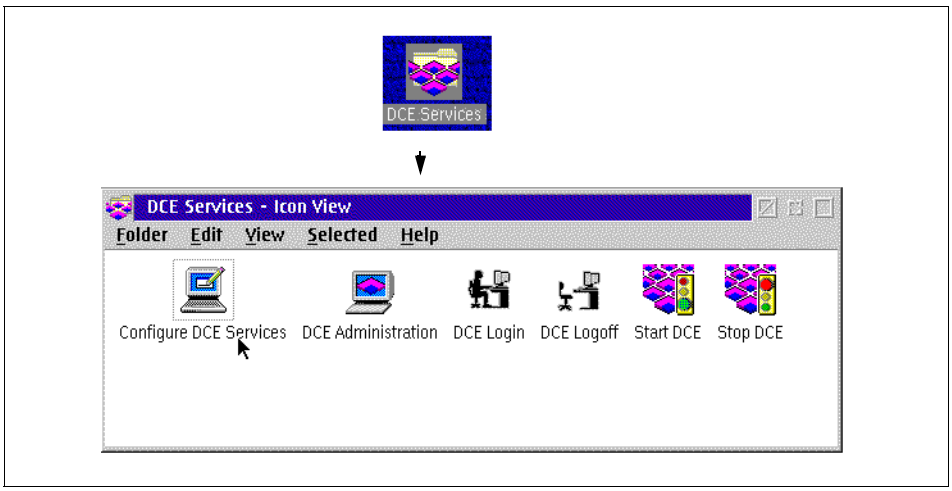


Figure 53. OS/2 Warp DCE Services

12. When the Distributed Computing Environment Configuration window (Figure 54) is displayed, check either the **Clients** or **Slim Client** options. The DCE slim client requires less disk and memory than the full DCE

clients and contains enough function to support GSO. If you choose to configure full DCE clients, you must select the **Security client** and **Directory client** on the DCE clients pop-up. The DTS client (Distributed Time Service) is optional but recommended to keep time synchronized across the cell. Click on the **Next** button to move to the DCE Configuration Catalog screen.

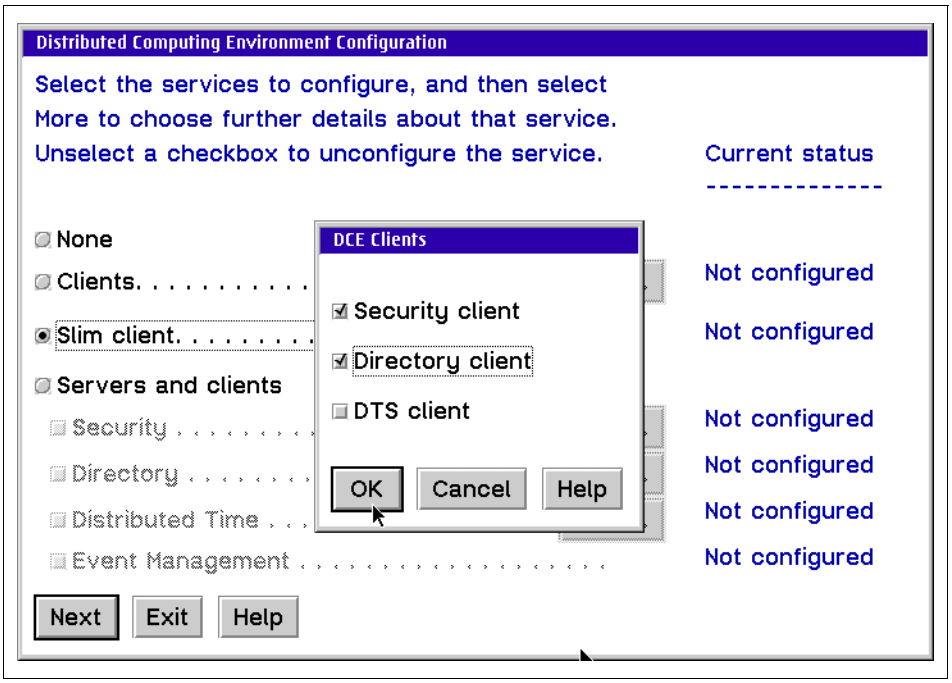


Figure 54. OS/2 Warp DCE Select Clients to Configure

13. You will find that you need to complete the **Cell Administrator** details. This account was created when the DCE server was configured. The default name is cell_admin. If this was changed during the server configuration, you will need to enter the actual cell administrator ID here. You will also need to know and enter the password for the cell administrator. Click on **Cell Administrator** and enter the cell administrator ID and password on the pop-up panel as shown below in Figure 55.

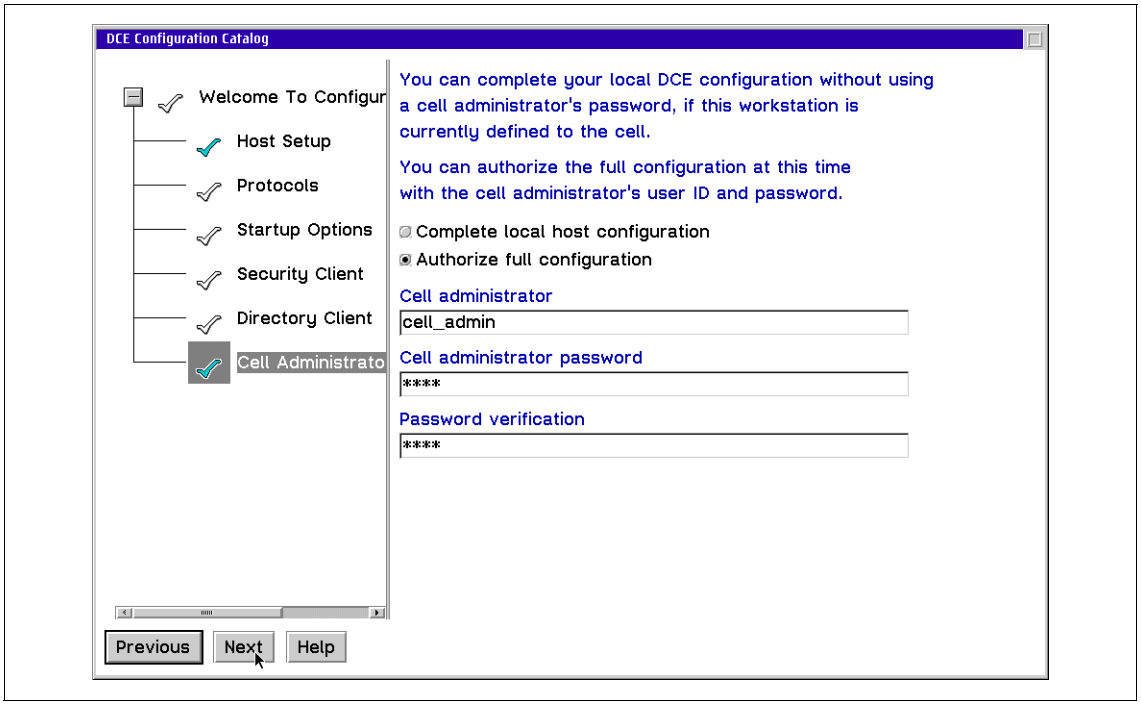


Figure 55. OS/2 Warp DCE Configuration - cell_admin

14. If you are configuring the full clients, you will also need to complete the Host setup details (Figure 56). You will need to complete the cell name and DCE host name. LAN Profile can normally be left unchanged. A common mistake at this point is to enter the host name of the primary DSS server in the DCE Host Name field. What is actually required is the hostname of the workstation being configured. Configuration will fail if the wrong hostname is entered.

15. The other details on the DCE Configuration Catalog screen can normally be left as is. Click **Next** from this panel to display the Confirm Configuration screen. If the list of components is correct, click **Run** to begin the configuration. If not, you can use the **Previous** option to go back and change options on the previous screens.

    Note that the DCE server(s) must be running for this configuration to succeed.

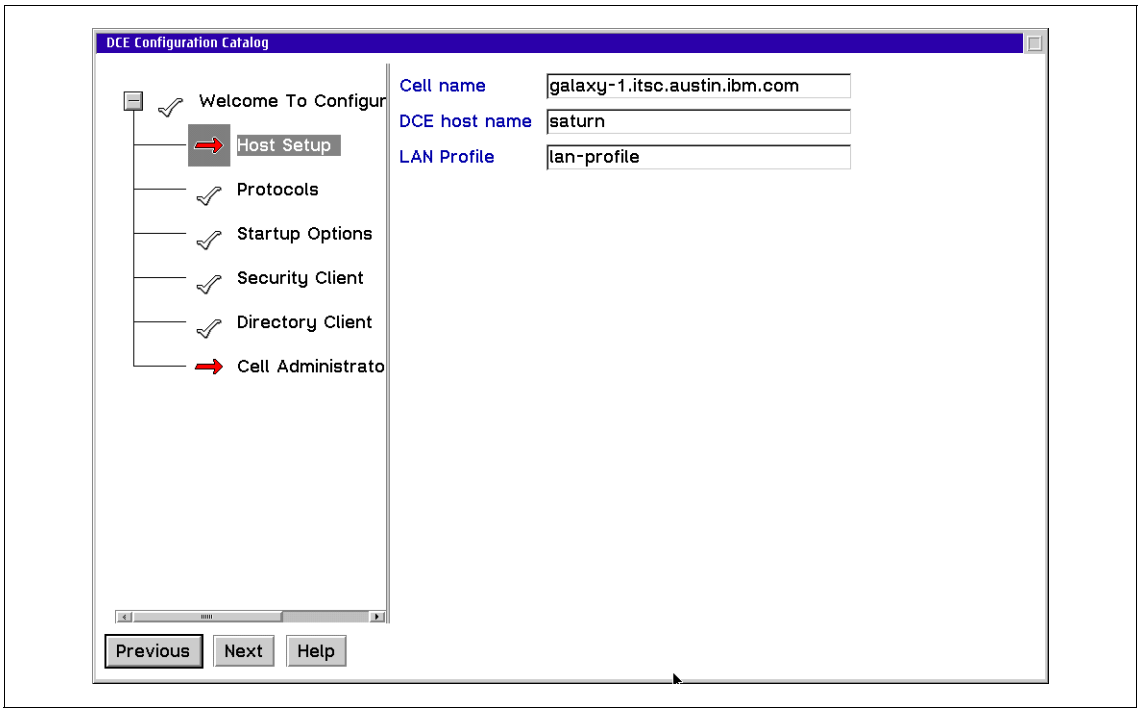


Figure 56. OS/2 Warp DCE Configuration - Host Details

16. A Configuration Progress screen will be displayed. Be aware that the security client is the first client to be configured. This can sometimes take a little while to complete. Do not panic and press **Stop** if the progress complete bar tends to stick at various points. The configuration process can take a few minutes.

If you do get configuration errors, you will need to reconfigure the client. To do this, follow the same steps as before. When you come to select the clients to be configured (Figure 54), you will notice that the current status will show as partial configuration or configured. This cannot be changed. You should instead select the same clients you selected previously, and when the configuration runs, DCE will unconfigure and reconfigure any selected clients.

As stated before, a common reason for the configuration to fail is time skew between the server and the client machines. Be aware that the first message displayed is `authentication failed for account cell-admin`; the `clock skew too great` message follows, and it is the real problem. You need to check and synchronize the date and time on the client machines and rerun the configuration. Sometimes this error causes a reconfiguration to stall at

around the 35 percent complete mark. This may be caused by some left-over information in the configuration files. If this happens, try to delete the following files:

```
c:\opt\dce_cf.db
c:\opt\dcelocal\etc\mclcfg.dat
```

and rerun the configuration.

If you have other configuration errors, you can look in the c:\opt\dcelocal\etc\cfgdce.log for possible causes. You should also consult the DCE online documentation supplied on the CD-ROM.

Once DCE has been installed, the IBM Global Sign-On for Multiplatforms, Version 1.5 client can be installed (note that the GSO OS/2 Warp client that ships with GSO 2.0 is actually at version index 1.5). The OS/2 DCE client must be installed before the GSO client can be installed, and it must be configured before the GSO client can be configured.

To install the GSO client for OS/2 Warp, do the following:

1. Insert the CD-ROM for the OS/2 Warp client into the CD-ROM drive.
2. From a command line, change to the \gso\<lang> directory on the CD-ROM drive and run **install**, where <lang> corresponds to the language designator for your language (for example, EN_US for US English).
3. After the welcome window, you will be asked whether or not the install process should update the config.sys file. It is recommended to have this file automatically updated rather than to update it manually afterwards.
4. The next panel lists the components to install (Figure 57). Select IBM Global Sign-On Product Files (which is the only selectable item) and specify an installation directory (or accept the default).
5. Click on **Install...** to start (and complete) the installation.

After the installation has completed, the workstation needs to be rebooted.

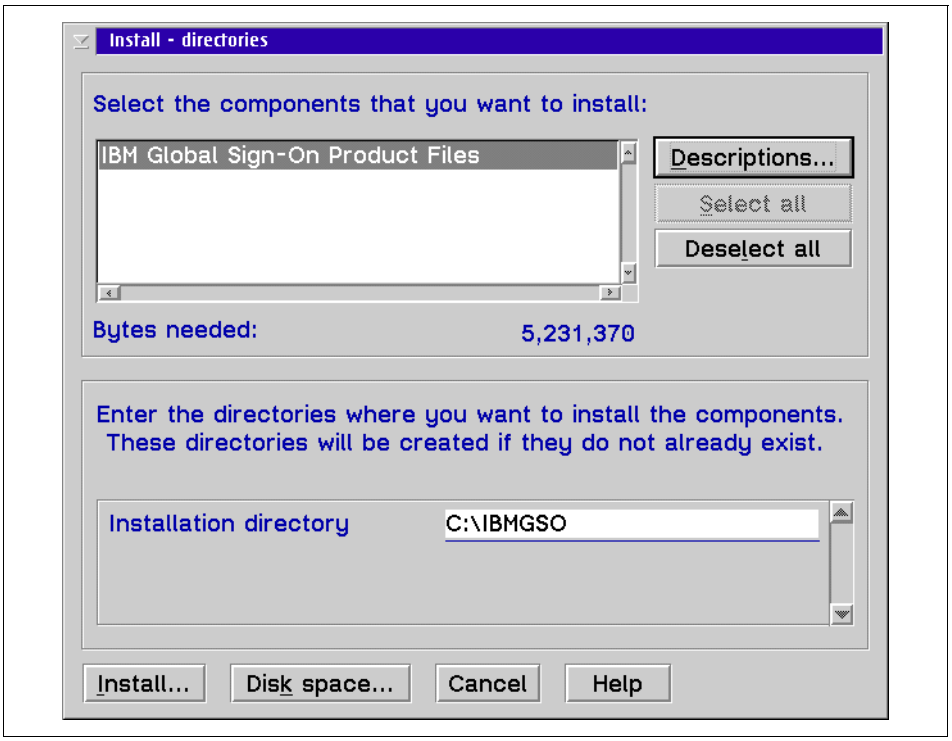


Figure 57. GSO Client on OS/2 Warp Installation Options

After rebooting, you will find that the GSO Launcher has been added to the startup program, and it will run every time the workstation is rebooted. Use **Cancel** to dismiss this dialog if DCE has not yet been configured or if you do not wish to log on to GSO.

There is no separate configuration steps needed for the GSO client. A **GSO Clients 1.5** icon is added to the desktop. If you double-click on this icon, you will find the **GSO Launcher** and **GSO Administration** icons that allow you to log on to GSO. You will also find icons for the help files and online documentation.

5.4 Adding Smartcard Support

If your environment uses Smartcards, GSO 2.0 gives you the option to use Smartcard authentication. Smartcard authentication is supported for logging on to GSO, which in turn logs you on to all the applications that have been configured as GSO targets.

There are two parts to the configuration of Smartcards for GSO on a user's workstation. The first part is to install and configure the necessary device support on the workstation to use the Smartcard device for authentication. Once this is done, GSO always requires that the Smartcard be inserted in the card reader and that a PIN (Personal Identification Number) has to be entered to unlock the Smartcard; it will not prompt for user ID and password. The second part of the configuration is to configure and initialize the actual Smartcard for the individual GSO user and to store his or her information on the Smartcard and in the security registry on the GSO server.

Note that once a user has been configured in GSO for Smartcard authentication, that user is always required to log on using a Smartcard. This provides for additional security. It means that disabling or removing a Smartcard device does not result in people being able to log on without a Smartcard. Logon attempts from any workstation that does not have a Smartcard reader will fail. This is true even if the Smartcard device has been unconfigured from the GSO client workstation and the GSO Launcher or Administration GUI prompt for user ID and password. If there is a need for a user to be able to log on to GSO without a Smartcard, for example from a location where the devices have not yet been installed, the user will need to have two user IDs, one for use with Smartcard and the other for use without the Smartcard.

If it becomes necessary, it is possible to unconfigure the client and reset the user ID to use the user ID and password authentication method. The user's password would have to be reset since the Smartcard support replaced the original password with a randomly generated password that the user would not know.

Global Sign-On for Multiplatforms, Version 2.0 supports Smartcards from Litronic, Inc. (www.litronic.com). Therefore, the configuration steps given here are provided for Smartcards from Litronic, Inc.

- The Litronic Smartcard reader and the Smartcard to be used have to be ready prior to GSO install and configuration. You should refer to the product documentation for information on how to install the device. You have the option of either initializing the Smartcard prior to GSO configuration, or you can initialize it while you are configuring GSO on the Smartcard. (Initialization of the Smartcard is a process where the Smartcard is authenticated, and the requisite data, like the user ID and password, are stored on it.)
- GSO installation for the Smartcard is done on the GSO client only. There is no configuration required on the GSO server.

- Once the user has been defined in GSO to use a Smartcard, that user cannot login to GSO until the user's Smartcard is inserted in the Smartcard reader and the user enters a valid PIN.
- Sometimes the Smartcard is not read by GSO while the GSO Smartcard administration client is invoked. When this happens, just remove the card and insert it again.

5.4.1 Installing the Smartcard Reader

Prior to using the Smartcard, the Smartcard reader has to be physically connected to the system, and the driver software has to be installed. For proper physical installation, please see the instructions that came with the reader. (Typical models need a serial connection to one of the workstation's serial ports and a power source from the keyboard connection.)

Then, run the installation and configuration program that was shipped with the Smartcard reader. It installs the necessary driver software and checks proper function of the reader and the Smartcard. You should not continue unless this tool proves that the Smartcard reader is properly working. It also allows you to change the security officer's PIN that is needed for subsequent storage and administration of GSO user information on that card.

Please refer to the documentation that came with the Smartcard reader for the latest or updated installation instructions.

5.4.2 Setting Up and Using GSO Smartcard Administration

What follows is a step-by-step description on how the GSO-related information can be stored on the Litronic Smartcard. Note that this does not need to be done on each client workstation; it will most likely be done on a security officer's workstation only.

1. On the workstation where GSO Smartcard administration will be performed, insert the GSO client CD-ROM in the CD-ROM drive.
2. Click on **Start** on the task bar of your Windows 95 or Windows NT system and click on **Run**.
3. Type in `x:\scadm\setup` and click on **OK**, where `x:` is the drive letter assigned to the CD-ROM where the GSO client CD-ROM is accessible. This initiates the installation process of the Smartcard administration software from GSO.
4. A Choose Setup Language window will now appear. Click on the language that you prefer to have the installation done in and click on **Next**. Skip the welcome panel by clicking on **Next**.

5. On the window that now appears, specify the installation directory (or accept the default) and continue to the next panel; it shows you the current installation settings for verification.
6. If you need to, click on the **Back** button to modify these settings. If these settings are correct, click on **Next** to start the installation process.
7. The files are then copied to the hard disk. After completion, you will get the Setup complete dialog. Click on **Finish** to close this window.

The previous steps described the installation of the Smartcard hardware and driver software as well as the GSO administration support for Smartcards on a security officer's workstation. At this time, the Smartcard needs to be initialized to support GSO for each user. It is assumed here that the Smartcard has not been used for GSO before, and it therefore needs to be initialized. Follow the steps below to accomplish this.

1. Click on **Start** on the task bar. Then click on **Programs** and click on **IBM GSO Smart Card Administration**. Now click on **IBM Global Sign-On Smart Card Administration**. This will launch a GSO login window.
2. You must log in with a user ID (and password) of a user who has GSO security officer authority since the following actions update the user information in the GSO server database. Logging in as a normal user would not be sufficient for this task. After successful login, the Smart Card for GSO dialog, shown in Figure 58, shows up.

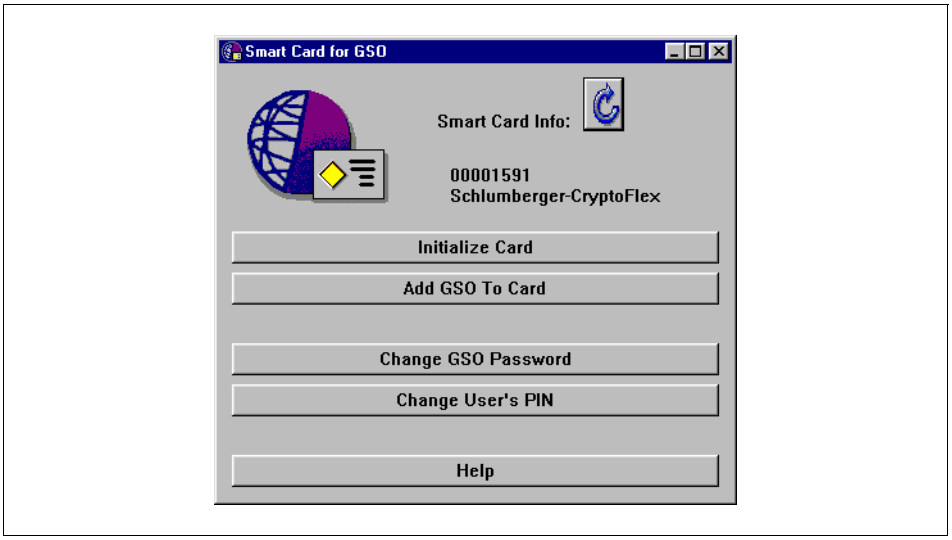


Figure 58. Smartcard Administration Window

3. Click on **Initialize Card**. Ensure that the card for the user you want to initialize is inserted in the Smartcard reader. This launches the Smart Card for GSO - Initialize Card dialog window (Figure 59).

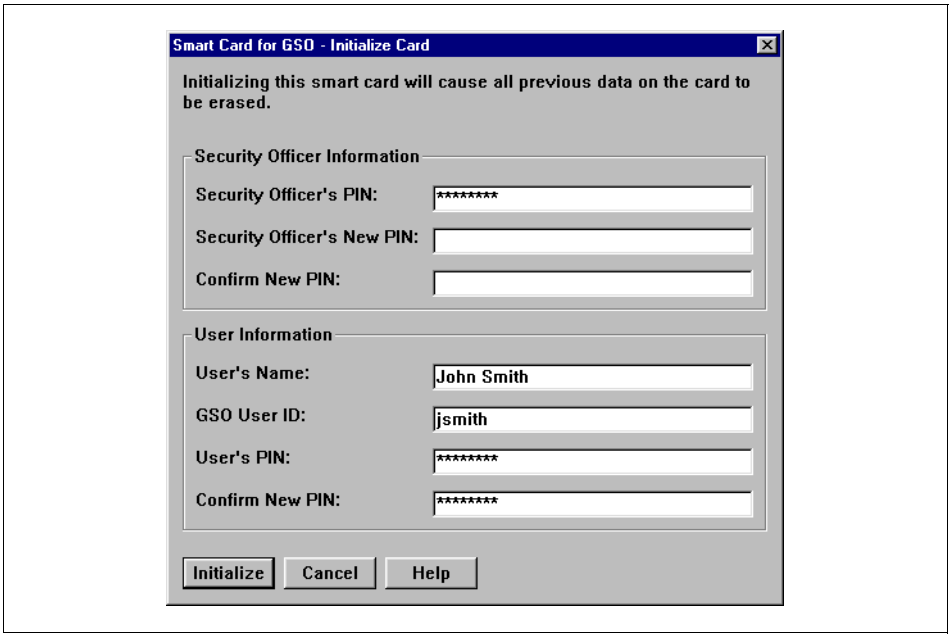


Figure 59. Initializing the Smartcard through GSO Smartcard Administration

4. Enter the security officer's PIN. This PIN is independent of GSO and is a security means to unlock the card for administrative purposes. For new cards, this PIN must have been provided with the card, or the previous administrator must provide it to you. Note that you have the opportunity to change the security officer's PIN on this dialog (Figure 59).
5. Enter the full name of the GSO user, the GSO user ID, and the user's PIN (twice for verification) in the fields below. Entering the user's full name may help identify the owner of a lost card.
6. After filling in this information, click on **Initialize** to continue and complete the process.

The user's information is stored on the Smartcard, and a random password is created for that user that is stored in the GSO server. The password will also be stored on the Smartcard. This (hidden) password is used for GSO authentication when the user logs on with the Smartcard; thus, the user does not need to know it.

5.4.3 GSO Client System Setup for Smartcards

The GSO client code on each GSO user workstation needs to be made aware of the Smartcard as an authentication method after the Smartcard hardware and the software drivers have been installed. The preferred way to do this is using the administration tasks provided by the Tivoli GSO Plus module, as follows (see also 7.1, “GSO Management Tasks” on page 205):

1. Start the Tivoli desktop as a user with sufficient administration authority.
2. On the Tivoli desktop, double-click on the **TivoliPlus** icon and then on the **GSO Plus** icon to open the GSO Plus collection window.
3. Double-click on the **GSO Administration Tasks** icon to launch the GSO administration task library window.
4. Locate the icon **Enable Litronic Smart Card** and double-click on it to bring up the Execute Task dialog window.
5. It is recommended to check the Display on Desktop output destination to be able to follow the process in a separate window that will be opened.
6. Change any options as required. At least, select the host(s) on which you want to enable the Smartcard from the Available Task Endpoints list. Make sure that the hostname(s) is (are) moved to the left list under Selected Task Endpoints.
7. Click on **Execute & Dismiss** (or **Execute**) to start the task. If you checked the Display on Desktop checkbox, a window opens that lets you follow the task and that indicates success or any problems of the execution.

Note that there is also a task that allows you to disable the Litronic Smartcard should that be required later on.

Enabling the Smartcard can also be done locally on the workstation. To do this, go to a command prompt and change to the GSO binary directory (by default c:\ibmgso\bin). Then, type the following command:

```
cfgclient -logindev -library c:\ibmgso\bin\litronsc.dll
```

Upon success, you get a message that shows that an authentication device has now been configured on this system.

You can use the `cfgclient` command to verify the configuration by typing:

```
cfgclient -view
```

The response from this command is shown in Figure 60. Notice that there is the litronsc.dll library file listed under Login device, indicating that the Litronic

Smartcard is being used for authentication (the Login device section would be empty otherwise).

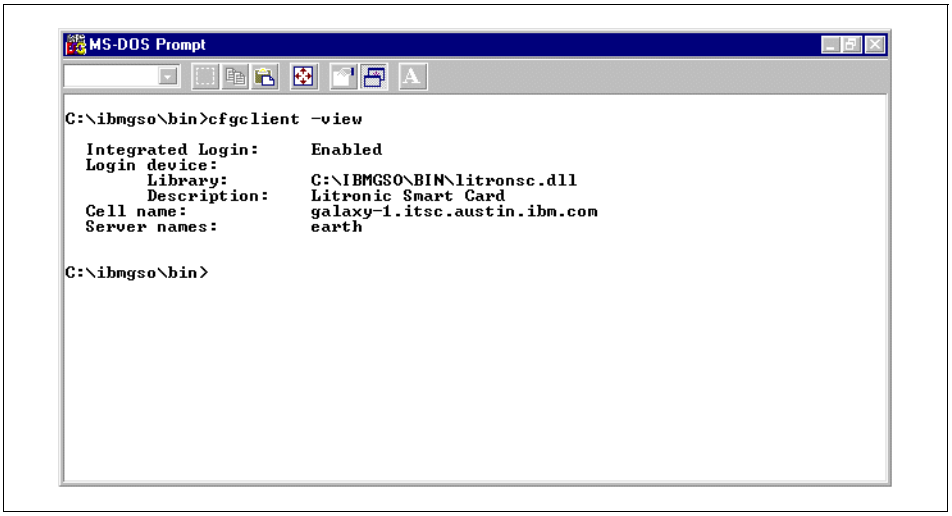


Figure 60. Checking Smartcard Support

To disable the use of the Smartcard reader with GSO on a workstation rather than with the appropriate Tivoli task, you can use the `cfgclient -logindev none` command. Remember, however, that the user's password needs to be reset before he or she can log on to GSO without a Smartcard.

This concludes the configuration for Litronic Smartcard support. When a user wants to log in to GSO, he or she has to insert his or her Smartcard in the Smartcard reader. GSO will then only prompt for the Smartcard PIN rather than for a user ID and password (Figure 61).

Figure 61. Sign-On with the Smartcard PIN

After entering the PIN and clicking on **OK**, the necessary user information is read from the Smartcard and passed on to the GSO server for authentication and to log the user on to GSO. When the user selects **Change Password...** from either the GSO Launcher or GSO Administration windows, a Smartcard

change PIN dialog will appear instead of a password change dialog that lets the user change his or her Smartcard PIN.

5.4.4 Miscellaneous Administration Tasks with Smartcards

As can be seen from the options shown in Figure 58 on page 133, there are some more functions available in regards to Smartcards. They are described below according to the buttons on that window.

Note: In order to launch the IBM Global Sign-On Smart Card Administration tool after that workstation has been enabled to use Smartcards, you must log on to GSO with a user ID that has GSO Security Officer authority. In real environments, this most likely requires a separate Smartcard for such an administrative user.

5.4.4.1 Add GSO to Card

This option allows an administrator to add GSO-related information to a Smartcard. If a user's information has already been stored on a Smartcard (that is, the card has already been initialized for use with GSO), this option allows you to change the user's name and the user ID on the card. Click on the **Add GSO To Card** button to initiate this function and change the information in the entry fields.

Note: Changing the user's ID should be done with care, and it requires that the new user ID already exists in the GSO database.

5.4.4.2 Change GSO Password

As explained in the previous sections, the basic authentication still uses a password between the GSO client and the server. This password is randomly created when the card is initialized for that user and stored on the Smartcard. The user, however, never sees that password because he or she never needs to use it.

If, for any reason, this password stored on the Smartcard gets out of sync with the copy stored in the GSO server, authentication for that user will fail. To correct this, the GSO Security Officer needs to choose the Change GSO Password function in the GSO Smartcard administration application. A window, shown in Figure 62, opens and requires the user's PIN for that Smartcard to be entered (the user's name, although shown, cannot be changed). Provided the user's PIN was entered correctly, clicking on the **Change** button creates a new (hidden) password and synchronizes the Smartcard with the GSO server.

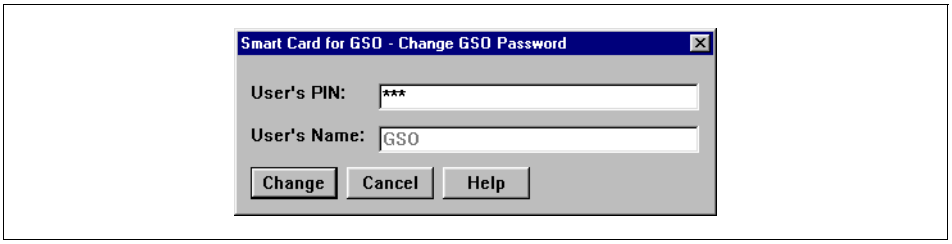


Figure 62. GSO Dialog for Changing Password

5.4.4.3 Change User's PIN

In cases where a user has forgotten his or her Smartcard PIN, the last function on the Smart Card for GSO panel lets an administrator define a new PIN. After clicking on the Change User's PIN button, the dialog shown in Figure 63 pops up. The security officer's PIN is required in the first field, and then a new user PIN can be entered on this dialog (twice for verification).

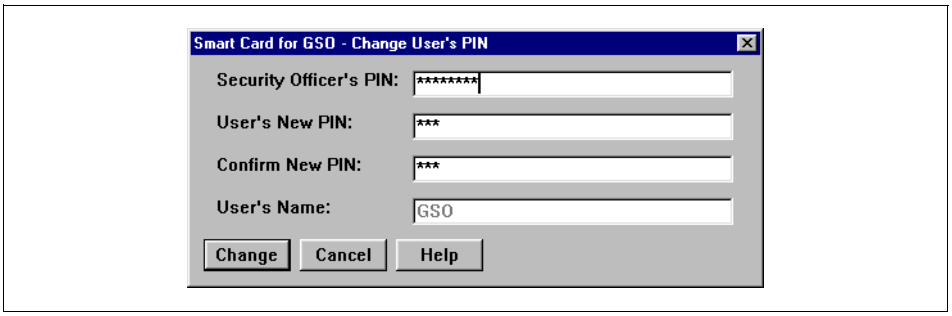


Figure 63. Dialog for Change User PIN through GSO

Note that the security officer's PIN represents an access key to be able to store the user's PIN on the Smartcard. The security officer's PIN has no relation to any GSO user or PIN.

5.5 Adding Biometric Support

IBM Global Sign-On for Multiplatforms, Version 2.0 supports the SecureTouch fingerprint reader from Biometric Access Corporation (BAC). Installation and configuration as described in this section is with reference to this device. Further details of the device can be found on the World Wide Web at `www.biometricaccess.com`. You should check this page for updates and support information.

Use of biometric devices differs somewhat from the use of Smartcards as described in Chapter 5.4, “Adding Smartcard Support” on page 130. With Smartcards, the user has to use his or her card to log in. With biometrics, the user has to use the fingerprint reader to log in if the device is in use at the place where the user is logging in. At other locations, where there is no fingerprint device installed, the user can continue to log in using his or her user ID and password. The use of biometrics is a growth industry, and the advent of devices such as the SecureTouch fingerprint reader will undoubtedly increase their usage. The requirement may change to be one of universal enforcement by user ID. Today, the main requirement is enforcement at device and location.

The SecureTouch fingerprint reader has to be installed and ready before it can be used with GSO. The users’ fingerprints also need to have been captured. You should refer to the product documentation for full details of how to install the device and enroll users. The following outlines the steps for your convenience. Support is available on Windows NT and Windows 95. The descriptions that follow are from a Windows NT installation, but they are not expected to be different on a Windows 95 system.

SecureTouch devices are available for parallel and serial connections. For a parallel connection, it is recommended to use the LPT1 port, although other parallel ports should work as well. For your reference, the system settings for an LPT1 port that worked with a parallel SecureTouch device were as listed in Table 21.

Table 21. Parallel Port Settings for Biometric Support

Port Property	**Setting**
Parallel Port	378h
Parallel Port Mode	Extended
Parallel Port Extended Mode	Bidirectional
Parallel Port Extended Mode DMA	No DMA
Parallel Port IRQ	IRQ 7

Note that these settings could obviously be different for other environments. Important, however, is that the parallel port is capable for bidirectional communication.

5.5.1 Installing the Device

The installation of the BAC SecureTouch fingerprint device on Windows NT and Windows 95 is very similar. An important difference, however, is that enrolling user finger prints is only supported on Windows NT. Windows 95 clients must therefore use a Windows NT server (preferably a domain controller for security reasons) to authenticate users using the BAC SecureTouch fingerprint reader. Depending on your individual setup, you may choose to enroll fingerprints and store the fingerprint data centrally on a domain controller only (for an increased level of security and ease of administration), or you might choose to do this on a user's client workstation.

Installing SecureTouch involves the following:

1. The device support software needs to be installed according to the manufacturer's instructions (usually by running `setup` from the driver diskette shipped with the fingerprint reader device). On Windows NT, you must be logged on as a user with administrative authority. When setup has completed, you will find that clicking **Start** -> **Programs** -> **BAC SecureTouch for WinNT** (or **BAC SecureTouch for Win95**, respectively) shows a number of different applications and the help files. The applications include The Fingerprint Lab which can be useful in trying out and becoming familiar with the actual device prior to configuring the working environment. To do the actual setup for the environment, however, you have to use the FP Manager from the control panel (Windows NT only).
2. The Authentication RPC Service, as well as the RPC Locator and RPC Service must be running on Windows NT servers that are used from Windows NT or Windows 95 clients for fingerprint authentication. To ensure this, you need to run the `fprpc -install` command from within the c:\winnt\system32 directory. Then, you should ensure that all three services have the startup option set to `automatic`. To check or set this, double-click on the **Services** icon from the Control Panel. Then, locate and select (click on) each of these services from the list and change the startup option to automatic if necessary. After changing, the machine needs to be rebooted.
3. After reboot, you will find that the NT logon panel has changed to the BAC logon panel. This shows the user ID and domain name. You should log on as an administrator to be ready to start setting up users. Change either entry as needed and click on **OK.** A second prompt will then appear asking for the password.

4. After you have logged on, double-click on the **FP Manager** icon in the Control Panel**.** The SecureTouch Finger Print Manager panel will be shown as in Figure 64 (the FP Manager is not available on Windows 95).

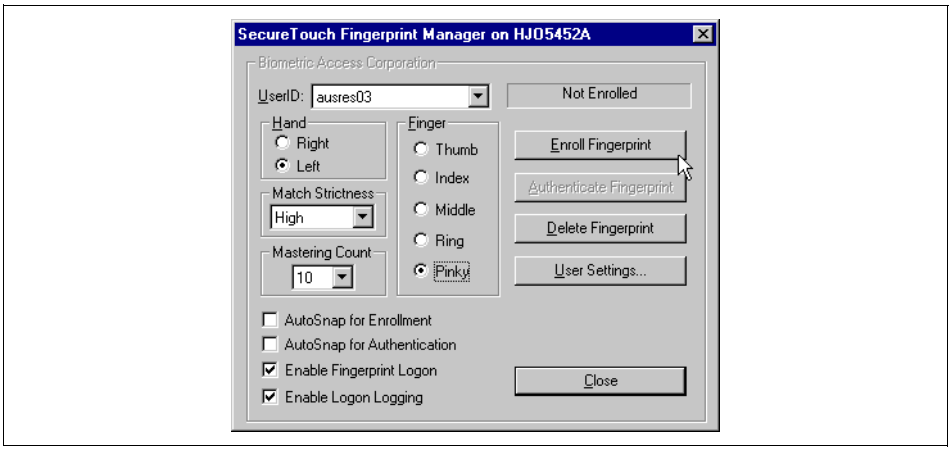


Figure 64. Enroll User Fingerprints

The user ID displayed is the user ID you logged on with. The top-right field shows that the user is not enrolled. Enrolled is the term used when a user's fingerprints have been captured. The advice is not to set up the Administrator user ID for fingerprint authentication. This will allow the Administrator ID to be able to log on without fingerprint authentication. This is useful in the event of a problem with the device and for remote installs. You should select another user ID. You then have the choice of which hand and which finger you would like to enroll.

5. The next thing to decide is what level of fingerprint checking you require. The GSO readme suggests that a mastering count of at least 10 be used together with a match strictness of high. These are actually the maximum settings you can have. What this means is that 10 separate captures of the finger will be made during enrollment. During authentication, points from all of these will be used for checking. This results in a high level of confidence.
6. The fingerprint reader can be set to autosnap, meaning to take a picture immediately when a finger is placed on the device. You may wish to turn this on for authentication in order to have the device react automatically at logon time. You should turn this off for enrollment. During enrollment, the BAC software checks each new picture of the same finger against the others it has taken. If you don't place your finger on the device in a similar

way as previously, you will be asked to do it again. The same is true if you suddenly switch fingers or change one with somebody else.

7. The Enable Fingerprint Login option shown on this screen is a systemwide logon option for Windows NT. If you do not select this option, then it will not be initiated at NT logon time, even if you select fingerprint login for an individual user. For this reason, you should check this box.
8. The **User Settings...** option (Figure 65) allows you to set individual logon options. This can be useful when you are in the process of setting up for SecureTouch by enrolling the users' fingers. You can choose to allow a user to log on with a password, with fingerprint or with both. The password entered is actually the GSO password that is used to log the user on to GSO.

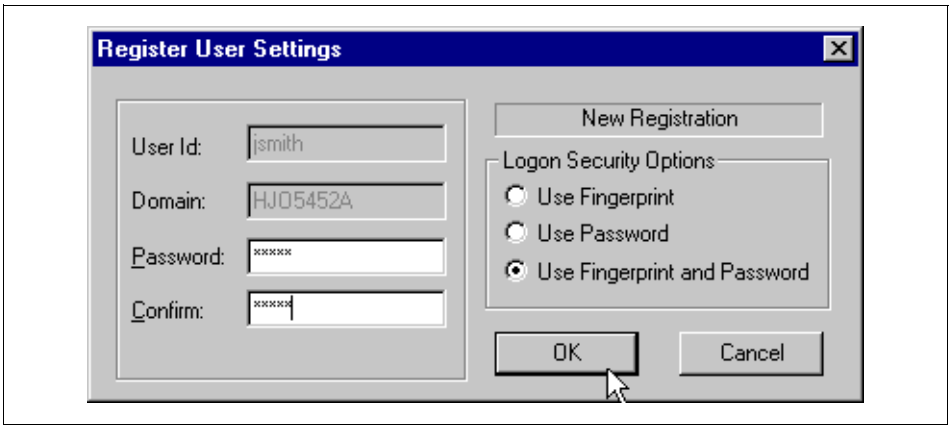


Figure 65. Finger Print Manager - Register User Settings

9. When you have completed all the options, click the **Enroll Fingerprint** option to start acquiring fingerprints. The Acquire a Live Fingerprint screen will be displayed as shown in Figure 66. You place your finger on the device and click **Acquire** when ready. This process is repeated as many times as the number set for the mastering count.

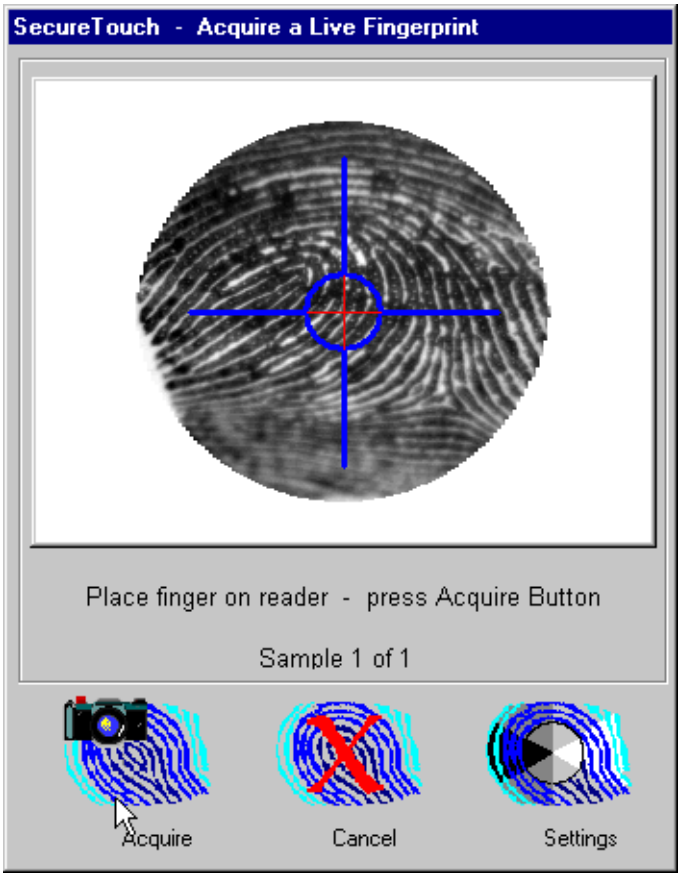


Figure 66. Finger Print Manager - Acquire Finger Print

10. When you have completed capturing one finger, you have the option to continue and capture more fingers. At the end of each enrollment, you can use the **Authenticate Fingerprint** option to check the finger you have enrolled. You should see the `Authentication Successful` message as shown in Figure 67.

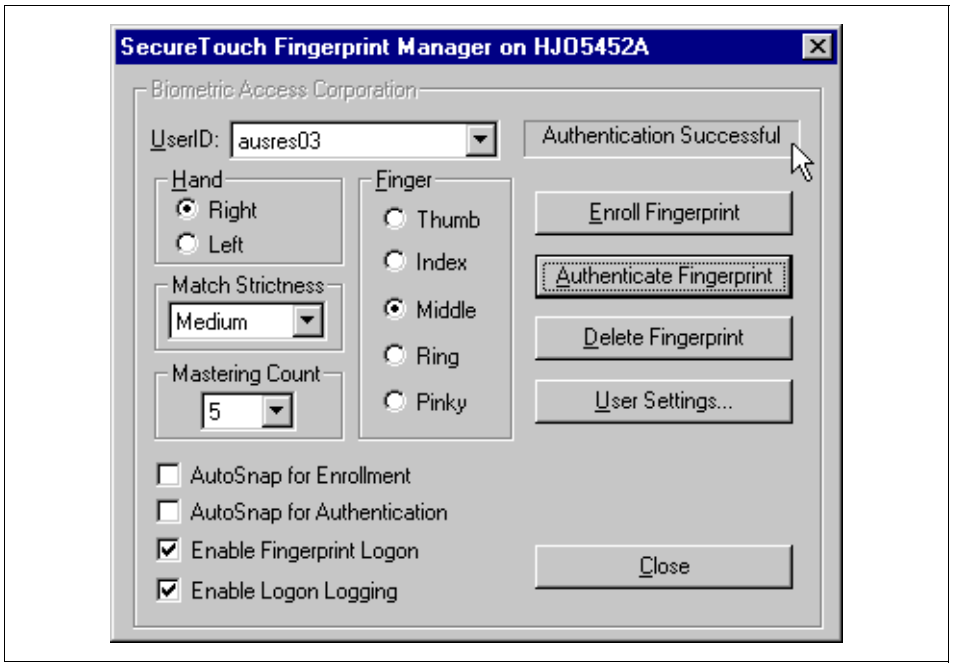


Figure 67. Finger Print Manager - Authenticated Fingerprint

You now have completed all the steps to enable fingerprints to be used for logon. If you log off and log on, you will find that after you have entered your password, SecureTouch will display an acquire fingerprint pop-up very similar to Figure 66. Place your finger on the reader to log on.

5.5.2 GSO Integration

There are two basic ways you can use the fingerprint support together with GSO.

If you have GSO installed with integrated logon enabled (which is the default), and SecureTouch is enabled for fingerprint logon (Windows NT only), it will pass the user ID and password to GSO for log on. You will need to have entered the password on the User Settings screen shown in Figure 65 on page 142. Using SecureTouch in this way may be what you wish for your environment. The workstation will be protected by the fingerprint logon, which requires the actual biometric data consisting of the live fingerprint and also a password.

The other option (the only option supporte on Windows 95) is to disable the BAC fingerprint logon for the workstation and enable the fingerprint logon for

GSO only. To enable GSO to use the BAC SecureTouch fingerprint reader, go to a command prompt and change to the directory where GSO binaries are installed (the default is c:\ibmgso\bin). Then type the following command:

```
cfgclient -logindev -library ibmbacfp.dll -server <auth server>
```

The authentication server (`<auth server>`) is the Windows NT server that performs the authentication. This authentication server needs to have the BAC SecureTouch software installed, and the required RPC services (as described under 5.5.1, “Installing the Device” on page 140) must be running.

Using the second option as described above (that is, *not* using integrated logon), be sure you disable the BAC fingerprint logon option (Windows NT only). If you do not do this, you will get two logon prompts for a fingerprint. The first will be from BAC, and the second will be from GSO. To disable the BAC prompt, go to the Finger Print Manager initial window, as shown in Figure 64 on page 141, and unclick the **Enable Fingerprint Logon** option. This disables the use of fingerprints at the initial NT logon.

If you log off from the machine and log on again, you will find that GSO prompts you for a fingerprint after the initial logon. GSO will again prompt for a fingerprint if you log off from GSO and then start either the GSO Launcher or the GSO Administration application later on.

5.5.3 Miscellaneous Tasks

If you wish to unconfigure SecureTouch from the GSO client, use the following `cfgclient` command:

```
cfgclient -logindev none
```

To remove the BAC SecureTouch software, you should use the **Uninstall** option under **Start** -> **Programs** -> **BAC SecureTouch For WinNt** (or **BAC SecureTouch for Win95**, respectively).

You should ensure that you have unconfigured BAC from the GSO client before you do this. If you do not, GSO will continue to prompt for a fingerprint and logon will fail.

If SecureTouch is removed from the system or if the device is removed, the administrator (or whichever IDs you chose) can still log on provided you did not set the fingerprint option.

Chapter 6. Defining Targets

After installing GSO server and GSO client systems, as described in the previous two chapters, configuring targets is the next step in deploying GSO. This chapter explains how to set up the various targets that GSO supports and outlines the special considerations for the individual targets.

Note

This redbook was written as a result of a residency project with limited time and resources. Due to these limitations, not all of the targets that GSO 2.0 supports could be explored and described herein. In particular, database and PeopleSoft targets are not covered in this book.

6.1 Target Systems

As described in Chapter 2, “Global Sign-On: The Macro View” on page 13, a target is a system or application for which GSO handles the end-user authentication functions for logon, logoff and changing passwords.

Authentication could be at the level of an operating system or on an individual application interface. In the rest of this chapter, the term *backend system* is used when referring to target systems in general.

In reality, the backend system could be local to the workstation where the GSO client is installed, or it could be on a separate machine elsewhere in the environment. The term is merely indicative of the fact that these systems are not part of the immediate GSO environment or under GSO control in terms of their being started and available.

There could be several targets running on the same machine. An illustration could be an OS/390 to which an end-user has separate logons for both TSO and CICS or an NT server in which the user is logged on to Lotus Notes and also to the NT Domain.

There could also be several targets in which the backend system is actually the same application. An example could be where an end-user has multiple sign ons to the same CICS region.

GSO is not in itself concerned with what the backend systems are or how many links there are to them. GSO is merely concerned with how to communicate with the backend system and what data to pass to the backend

system authentication interfaces. It supports communication through API, CLI and EHLLAPI scripts.

6.1.1 Target System Communication

It's important to understand that GSO does not provide the communication method. It does not, for example, provide emulation programs. It also does not replace an executable supplied to invoke an application, for example the Lotus Notes executable, notes.exe.

For some of the targets supplied with GSO, code is provided to interface with the backend system. It is, of course, this code which is defined as the thing to be called or invoked to perform various functions, such as logon. However, in defining a target to GSO, you are, in effect, defining what has to be invoked to automate the mechanics of communicating to a backend system. GSO may provide code to handle messages, responses and so forth, but this GSO code will still be calling the relevant product interfaces.

As an example, if in your environment PCOMM (IBM e-Network Personal Communications for Windows NT and 95) is used to communicate with host systems, the definition in GSO will be to use PCOMM. The actual definition points to the supplied GSO interface (or *wrapper*) code, gso3pcm.exe. However, this GSO-supplied program works to the interfaces supplied by the PCOMM product. GSO does not change PCOMM or enhance its functionality. It uses available functions as provided. PCOMM must obviously be installed and working outside of any GSO install and configuration. Additionally, the session that is to be used to communicate with the host system must have been defined to PCOMM prior to being used by GSO.

The requirement to have the ability to talk to the backend systems before using GSO to communicate with them is common to all targets.

GSO uses the term *program* when referring to communication methods. Whenever you define a target, you have to define the *program* to be used. These definitions are made using a *Program Template File (PTF)* and are held in the GSO *Program Database.* How to define target programs and how the PTFs and program database functions is covered in detail later in this chapter.

6.1.2 Authentication to Target Systems

The way that GSO works is that the users authenticate to GSO, and GSO then automates the logging on of users to their backend systems. To do this successfully, the user must have already been defined to the appropriate backend security systems. This can be done either natively or, where

supported, using Tivoli. Tivoli supports defining users to UNIX, NT, NetWare, RACF, and others. In the same way as GSO utilizes existing communication methods for backend targets, it also utilizes existing security methods. GSO does not require any changes to be made to the backend security systems.

The user obviously needs to be defined to GSO. This is done through TME 10 GSO User Administration (a GSO add-on to the Tivoli User Administration application) by way of a user profile. This definition sets up the primary or initial authentication that the user has to make to sign on to GSO, be it through user ID/password, Smartcard or biometrics. The targets the user is authorized to use are also defined through TME 10 GSO User Administration, using the same user profile. Target data that is held with the user definition is primarily that data needed to log this particular user onto a particular target. In other words, it is the user ID and password specific to this user that needs to be passed to the backend security manager. There can also be up to three target-specific entry fields associated with a user target. These are used related to a specific instance of a target type to a particular backend system; an example would be a host name for a 3270 target.

The target types that are available within a GSO cell are determined from the GSO *schema* files. The *schema* also defines the target-specific fields that need to be entered as part of the target definition.

Schema files are held at the GSO server. The targets supplied with GSO are added to the Tivoli Management Environment (TME) when the GSO cell is installed and configured, meaning supplied target types are already available to be defined for users through the user profile. New schemas only need to be defined, and the GSO cell information updated, if you add new targets. Section 6.7, “Adding New Targets to the GSO Framework” on page 199, covers how to do this and provides more detail on the relationship between the PTFs and the schema file.

When the user profiles have been set up, they need to be distributed from TME to the GSO server. The initial (or primary) authentication to GSO uses the DCE authentication mechanism. The GSO server defines users to a DCE security registry. Information relating to a users targets is also held, fully encrypted, in the DCE security registry as Extended Registry Attributes (ERAs).

Runtime options allow a user to change his/her passwords on target systems. These changed passwords get stored with the user target data in the security registry on the GSO server. Once a password has been changed, the user no longer needs to know it because GSO will perform the logon for the backend

system. The user should only need to be concerned with his/her single, primary logon.

User definitions, target definitions and runtime options are covered in further detail throughout this chapter.

6.1.3 Target Component Parts

The following diagram (Figure 68) illustrates the various pieces needed to define a target to GSO, where they are used, where the resultant data is stored, and how they are linked together. Detailed descriptions of the PTFs and so forth can be found in 6.7, “Adding New Targets to the GSO Framework” on page 199. The diagram here is to aid understanding of what is being used and/or updated within the GSO environment as you go through the process of adding targets for users.

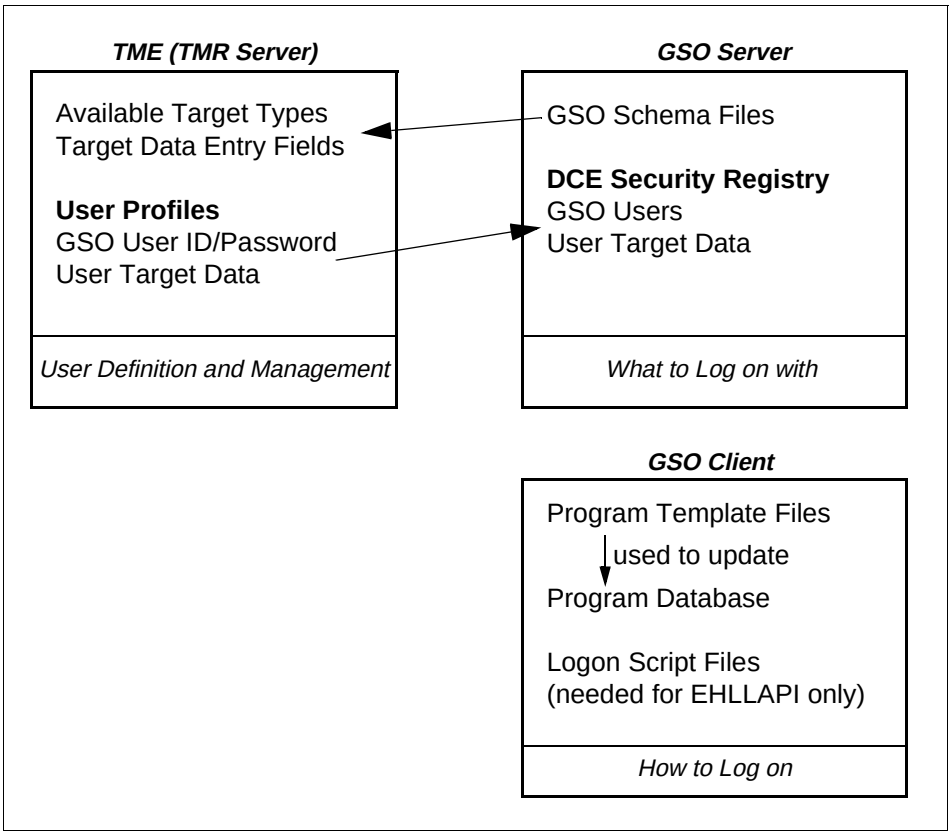


Figure 68. GSO Data Objects

The following diagram (Figure 69) shows a simplified GSO logon flow (see also 2.2.2, “The Details” on page 17). The user signs on to the operating system and to GSO and authentication is done through DCE. The GSO Logon Coordinator (LC) is then initiated at the client machine. The Logon Coordinator uses information from the Personal Key Manager (PKM) and from the Configuration Information Manager (CIM) to launch the backend targets. PKM extracts user target information from the security registry. This data is transferred through authenticated RPCs encrypted at the packet-privacy level. The data is not stored at the client. Configuration information stored at the GSO client is the data in the PTFs, in the program database and any script files.

As we will see, there are various options for when targets are actually started. They can be logged on when the user logs on to GSO, or they can be started by the user clicking on the logon option later. The Logon Coordinator process outlined is the same regardless of when the target is logged on.

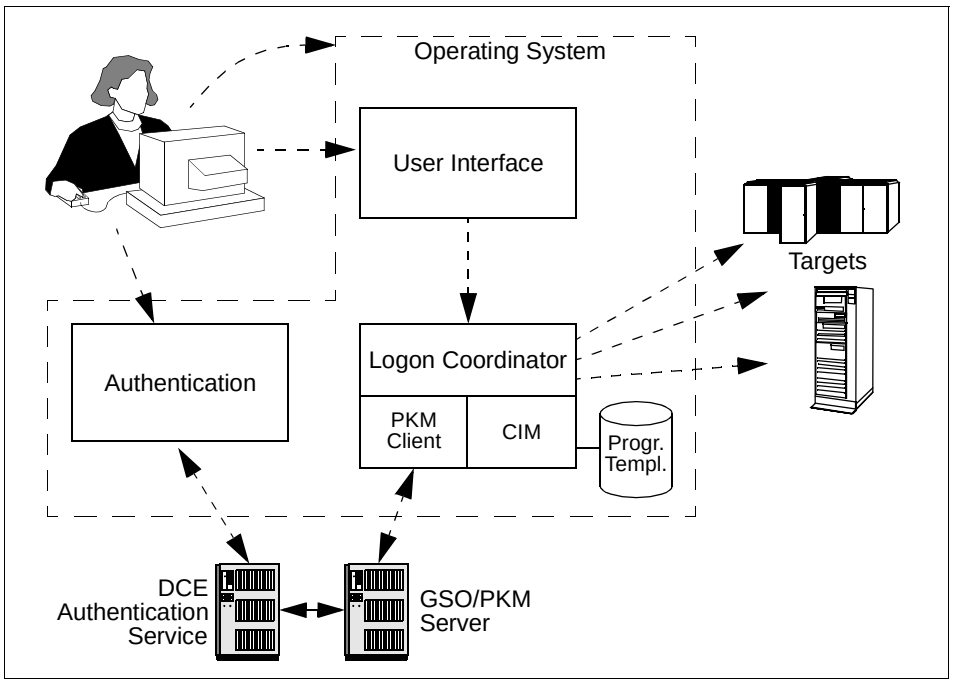


Figure 69. GSO Logon Flow

6.2 Adding GSO Programs

GSO targets are objects that describe *what* a user wishes to logon to. For example, a 3270 emulation GSO target object will have a unique name, user ID, password, host name, and an optional application name. These targets objects are stored on the GSO server.

GSO programs are objects that describe *how* the logon will be accomplished (as well as the logoff and change password). For example, a 3270 emulation GSO program object will contain a unique name, indication of which emulator to use, the location of the Logon Script File (LSF), time-out values, and so forth. The program objects are stored on each GSO client.

During target launching, the GSO client will retrieve a user's GSO target information from the GSO server, match each target with the appropriate GSO program, and the combination of the two will be used to log on to the target application. The match is done in either of two ways:

- The target object specifically names the program to use.
- The target object does not indicate the program to use; so the first program object that has the same Target_Type identifier as the target object will be used.

The mechanics of adding programs is the same regardless of the type of target that it will support. This section explains how to add, update, and remove programs, but does not cover any particular entries needed for different target types. If you are adding programs, you should review this section and then look to section 6.4, "Supplied Targets" on page 171, for an explanation of any target-specific fields or setup required.

The GSO program database is located in the <ibmgso>\config directory and is initially empty. A program is created by customizing a Program Template File (PTF), and the resulting object is added to the database.

PTFs are kept on the client side and not on the server side. GSO provides a set of PTFs that support various standard target applications. They are stored on every GSO client under the <ibmgso>\template directory. If desired, these templates could be installed on a central machine and accessed through a shared drive rather than be kept on every client. In an environment where the desktops and software installs are standardized, for example Lotus Notes is always installed to the C: drive, you might also want to consider having a central set of definitions and using Tivoli Software Distribution to distribute them as required.

You should not change the supplied Program Template Files or add new templates under the <ibmgso>\template directory. The files may be overwritten if the product is reinstalled or updated.

The default program names supplied with GSO are rather long. Since some GSO targets do require a specific program name to be entered on the TME Add Target panel, and the TME panel does not provide a list of the available program names, it is suggested that you change the default program name to something shorter and easier to remember by the TME administrator. Only two of the target types supplied with GSO require a program name: 3270 emulation and 5250 emulation. The remaining target types do not require a program name since they will be matched at launch time using the common Target_Type identifier.

Programs are added through the Global Sign-On Administration GUI. If this has not been added as a separate icon to the desktop, you can find it from **Start** -> **Programs** -> **IBM Global Sign-On Client V2.0**. When launched, you are presented with a GSO sign on panel; if you are already signed on to GSO, you will be asked, before this panel is presented, if you wish to continue using the same user ID or sign on as another user. There is no restriction within GSO as to who can use this GUI. Users only have to be defined to GSO; they do not need to be defined at the workstation. If you do not wish all users to be able to add programs, you will have to use local security to prevent this and/or disable or remove the Administration GUI.

The following steps outline how to manage program entries.

1. Sign on to GSO using the Global Sign-On Administration GUI. The sign-on screen is the same as the standard GSO Sign-On pop-up. You enter your GSO user ID and password.
2. The Administration window will be displayed. The only administration options available with IBM Global Sign-On for Multiplatforms, Version 2.0 are to view, add, update, remove, or refresh programs, or to browse the main GSO log. The first time you invoke the administration function, the program list will show as blank. To add programs, you should click on **Add** from the **Programs** pull-down menu (you could use the “+” icon from the icon bar, too), see Figure 70.

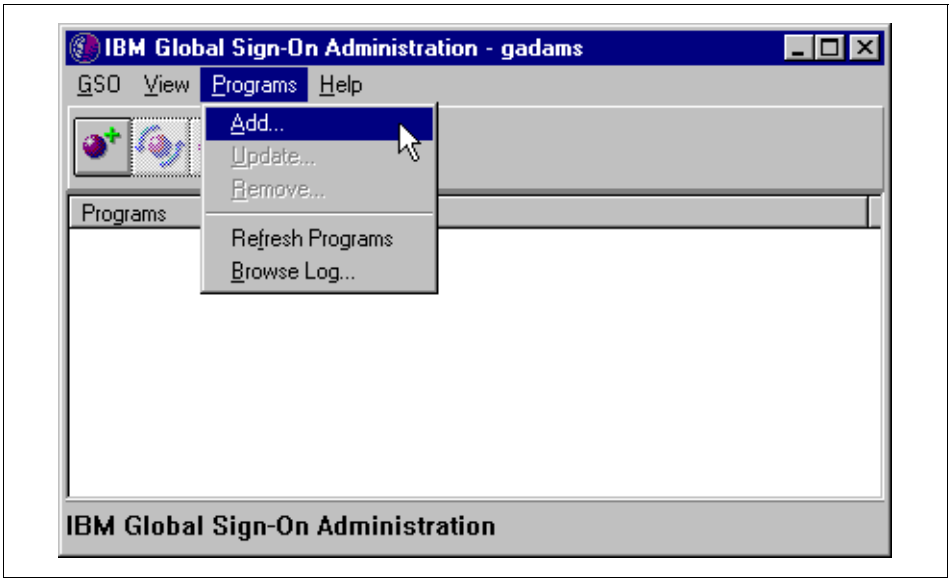


Figure 70. Global Sign-On Administration

The Global Sign-On Add Program window will be displayed (Figure 71). The top part of the panel shows the location (drive and directory) of the template files; this location is the default install location for GSO. If you use a different location, you need to overtype this field. You should then click on **Get List** to refresh the template list with the program templates available in that directory. You need to click on the name of the template you wish to use. The program name and location will automatically complete with the details from the program template. As you can see from the screen (Figure 71), the names of the programs for the targets supplied with GSO are extremely long. If you wish to simplify the program name, you can do this by overtyping it in the name field.

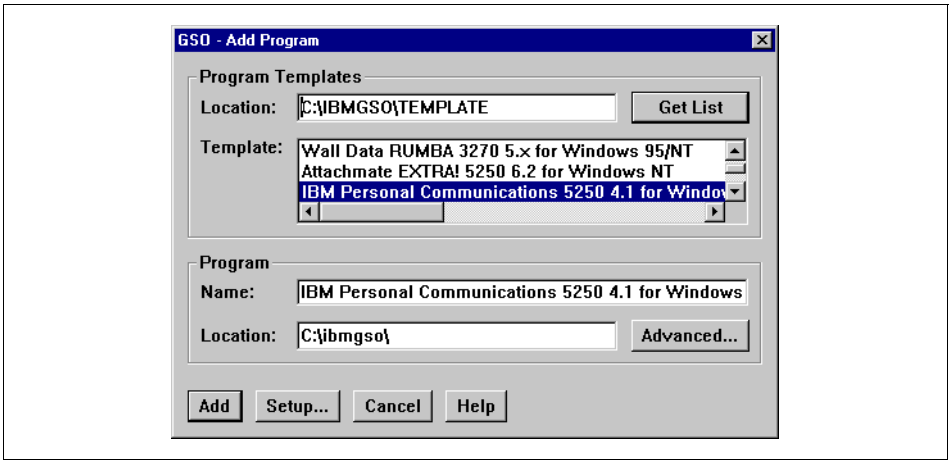


Figure 71. GSO Add Program

3. If you click on **Advanced**, the Advanced Program Path Configuration dialog window (Figure 72) will be displayed. This shows the path for the executables to be invoked to perform the various functions. Again, the initial definitions are those in the Program Template File. Only those functions for which there is an entry in the program template will be highlighted. In the example shown in Figure 72, which is for IBM PCOMM, there are entries for all functions except **Start**. If the location of the executable is different from that shown, you should overtype the field with the correct location. You can use the **Browse** button to look at local disks to check for locations, if required. When all the fields have been completed, click **OK** to return to the main GSO Add Program screen.

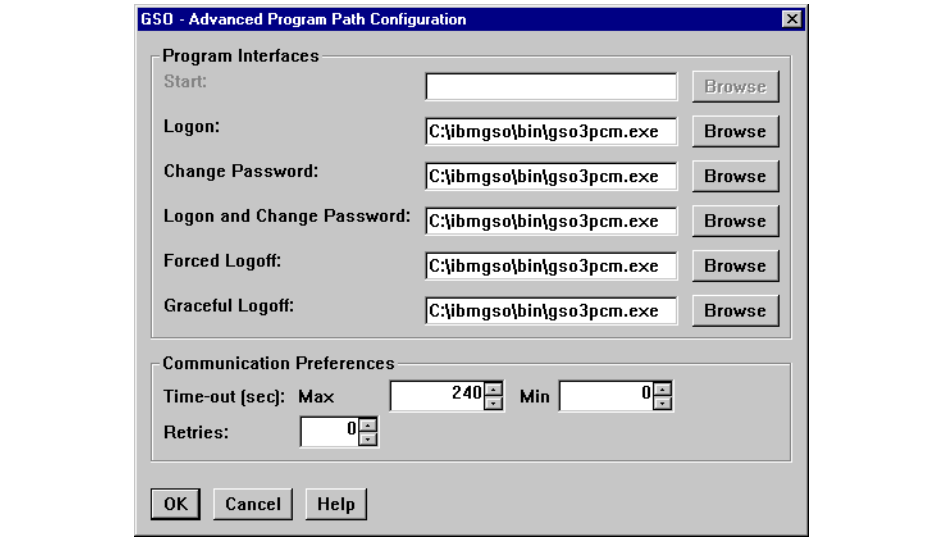


Figure 72. Advanced Program Path Configuration

4. When you return to the main GSO Add Program screen, you may have the option to perform additional setup. This option is only available (highlighted) if there are additional variables defined for the target type in the program template file. In our IBM PCOMM example, the **Setup...** button is highlighted. If you click on this, the following GSO Program Setup window is displayed (Figure 73). In this case, the additional fields are Session Profile, Session ID and Script File Path. Other target types will have different program setup values, or may not have program setup values at all. If the setup values shown are incorrect, you should overtype them. When all the values have been entered correctly, click **OK** to return to the main GSO Add Program window.

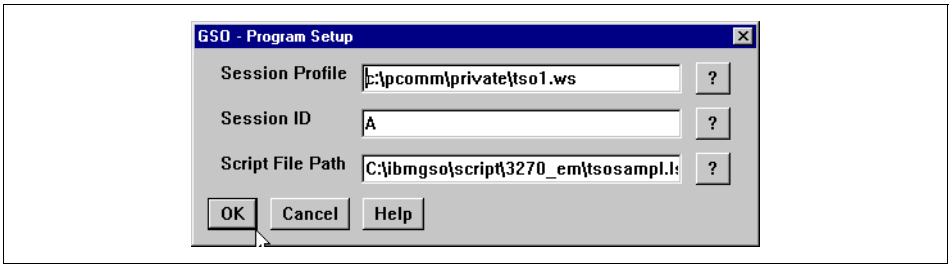


Figure 73. GSO Program Setup

5. From the GSO Add Program window, you should click **Add** to add the program. This action causes an entry to be made in the program database, <ibmgso>\config\pgm.db. Values from the program database are substituted at runtime for the values in the PTF.
6. As programs are added, they appear in the list on the main GSO Administration window (Figure 74). The programs appear with the names entered during the program add operations. As shown below, most program names were changed to simpler names.

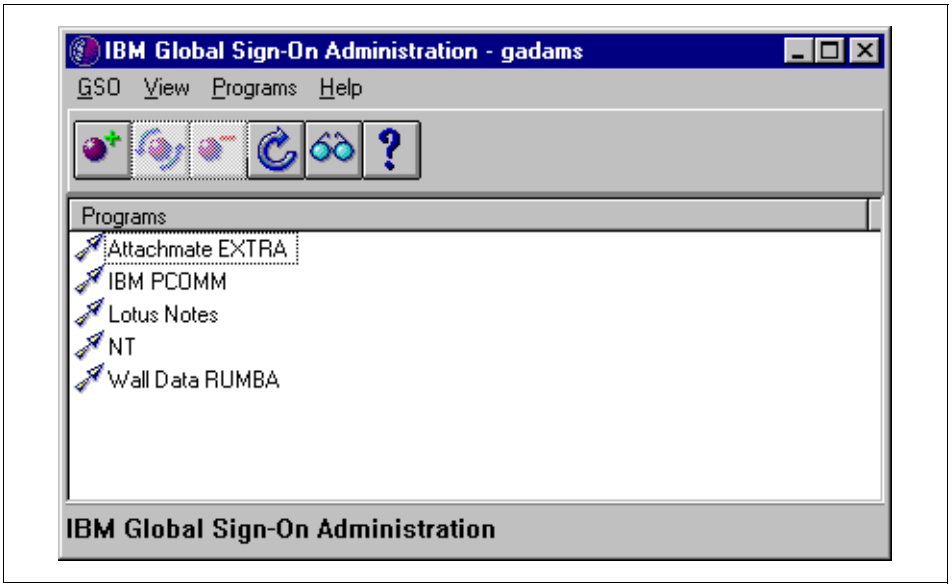


Figure 74. GSO Administration - Added Programs

7. Once programs have been added, the functions to update or remove programs become available. To use the update or remove function, you need to highlight the program you wish to change or delete and then use the appropriate function from the **Programs** pull-down menu or the specific icon from the icon bar (see example shown in Figure 75).

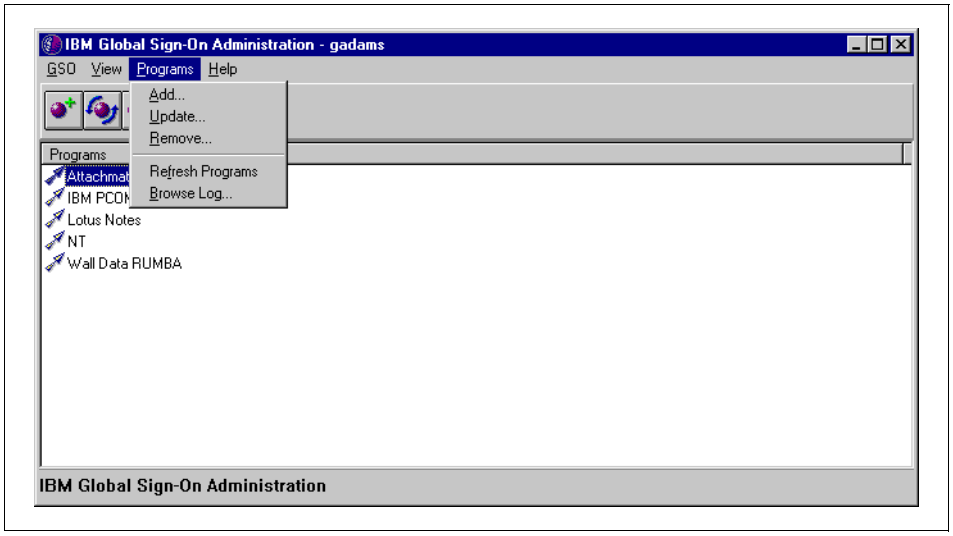


Figure 75. Update/Remove Program Options

8. The Update program function opens an initial program screen (Figure 76) very similar to the Add Program screen. The difference is that the program templates are not shown. If the change you wish to make is to use a different program template, you will need to remove the program and re-add it. You can, if you wish, change the name you previously gave the program. The advanced and setup options are as described in the sections for adding programs above. You may click on either of these and make changes as needed. When all changes have been made, you should click on **Update** to update the program database.

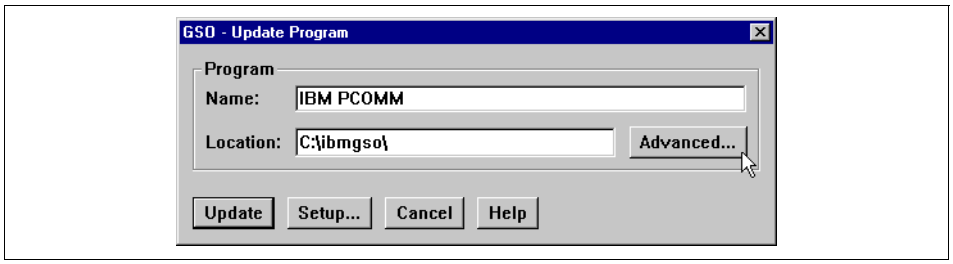


Figure 76. GSO Update Program

9. When you select to remove a program, you will get a pop-up asking you to confirm that you wish to remove the program from the database. You should note that the remove function will only remove a program entry

from the program database; it will not remove the program template. The template remains available and can be used to re-add the program if required again later. The remove action will succeed even if there is not an entry in the program database; its action is purely to ensure there is no entry in the database matching this program name and to remove the entry from the program list shown on the administration window.

Figure 77. GSO Removing Programs

10. As mentioned at the beginning of this section, you should not change the supplied Program Template Files. You should also not store new templates in the <ibmgso>\template directory because this directory may get over-written if the product is reinstalled or updated. The best advice is to use the add program functions to make necessary changes for supplied programs and have a separate directory for any new templates you develop. If you ignore this advice, however, perhaps for the expediency of making changes directly to the templates rather than through adding the programs, you should make sure you do the following:

    1. Save the contents of the supplied <ibmgso>\template directory to a safe place and ensure you know where it is. It is always a good idea to save the original files before making changes to them.
    2. Ensure you have a backup of your new templates to cover the eventuality of installing a new version of the product or reinstalling the existing version.

It can be extremely useful to look at the Program Template File for any of the supplied targets you will be using. The PTF often contains information in addition to the Readme files or the product documentation. For your

convenience and reference, some PTFs shipped with the product are listed in Appendix B, “Program Template Files” on page 255. This is for the convenience of people reading this book who do not have ready access to an installed GSO client system. Obviously, the PTFs are subject to change, and you should review an actual installed system for the latest content. If you wish to understand more about the contents of PTFs and how they are coded, this is covered in Chapter 6.7, “Adding New Targets to the GSO Framework” on page 199.

The PTFs available varies by client platform. This can be an indication that a supported target is not available for a particular platform. While this is usually true, it is not the case with respect to logging on to LAN Server from a Windows NT client. The IBM LAN Requester PTFs are not available on Windows NT clients. However, it is still possible to log onto a LAN Server target by using the Windows NT 4.0 target type. Conversely, you can use the LAN Server Logon to a domain target type to log on to NT from OS/2 Warp. You should review 6.4, “Supplied Targets” on page 171, for information on which targets can be launched from which clients.

6.2.1 Available PTFs by Client Platform

Program templates available on Windows NT clients are:

- Attachmate EXTRA! 3270 6.2 for Windows NT
- IBM Personal Communications 3270 4.1 for Windows 95/NT
- Wall Data RUMBA 3270 5.x for Windows 95/NT
- Attachmate EXTRA! 5250 6.2 for Windows NT
- IBM Personal Communications 5250 4.1 for Windows 95/NT
- IBM Client Access/400 for Windows 95/NT
- Novell NetWare Server (Bindery) for Windows NT
- Novell NetWare NDS for Windows NT
- Lotus Notes 4.X
- Windows NT 4.0
- SnareWorks V2.0

Program templates available on Windows 95 are:

- Attachmate EXTRA! 3270 6.2 for Windows 95
- IBM Personal Communications 3270 4.1 for Windows 95/NT
- Wall Data RUMBA 3270 5.x for Windows 95/NT
- Attachmate EXTRA! 5250 6.2 for Windows 95
- IBM Personal Communications 5250 4.1 for Windows 95/NT
- IBM Client Access/400 for Windows 95/NT
- IBM LAN Requester manage password in a domain
- Novell NetWare Server (Bindery) for Windows 95

- Novell NetWare NDS for Windows 95
- Lotus Notes 4.X
- Windows NT 4.0
- SnareWorks V2.0

Program templates available on OS/2 Warp are:

- IBM Personal Communications 3270 4.1 for OS/2
- IBM LAN Requester logon with domain verification
- IBM LAN Requester logon with local verification
- IBM LAN Requester manage password in a domain
- IBM LAN Requester manage password on a server
- Novell NetWare Server (Bindery) for OS/2
- Novell NetWare NDS for OS/2
- Lotus Notes 4.X

6.3 Adding Targets for Users

Targets are added for users through TME 10 GSO User Administration. The process is the same regardless of the target type. This section explains how to add, edit, or remove targets, but does not cover any particular entries needed for different target types. If you are adding targets, you should review this section and then look to the section under 6.4, “Supplied Targets” on page 171, for an explanation of any target-specific fields or setup required. The section is written with targets being added to users who have already been defined. It is obviously possible to define the user and add their targets in one step because the process, with respect to target information, is exactly the same.

1. From the Tivoli desktop, open (double-click on) the policy region in which the profile manager resides that holds the user profile(s) that you want to work with. Double-click on that profile manager to open the profile manager window and then double-click on the user profile that contains a particular user’s properties. This opens the User Profile Properties window as shown in Figure 78.

   (User profiles that store users’ properties are contained in profile managers in the Tivoli Management Environment. Profile managers, in turn, belong to a policy region. The way policy regions, profile managers and user profiles are arranged completely depend on the specific setup in your environment. There might be just a single instance of each in a small environment, or there might be many policy regions, profile managers and even more than one user profile within each profile manager. Therefore, we assume that an administrator knows where to find a particular user’s properties.)

2. Select the user you wish to add targets for from the User Profile Properties list and click **Edit User...**.

   Note that the user list in the User Profile Properties window has a horizontal scroll bar. By scrolling to the far right, the GSO properties for each user in the list could already be seen.

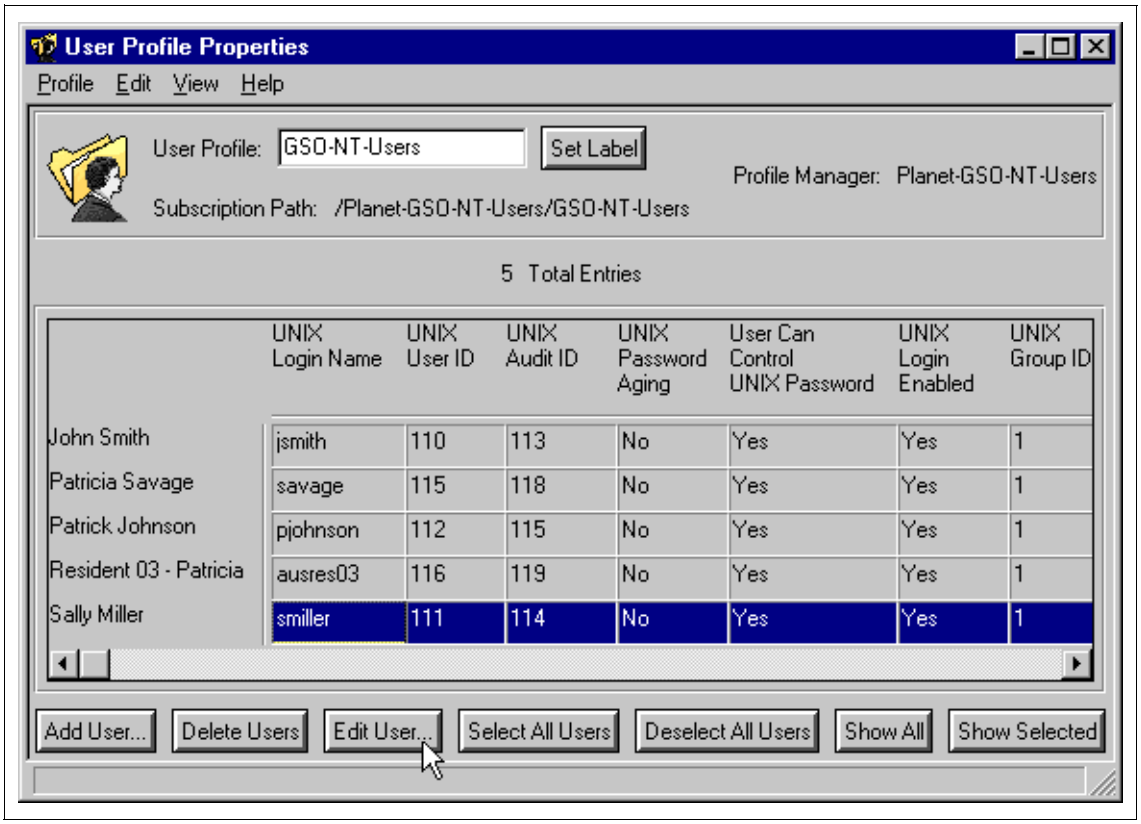


Figure 78. Selecting User Profile

3. When the User Properties window is shown (Figure 79), you should change the **Category** field to **GSO**. The user record will then show the entries for that user that are specific to GSO.

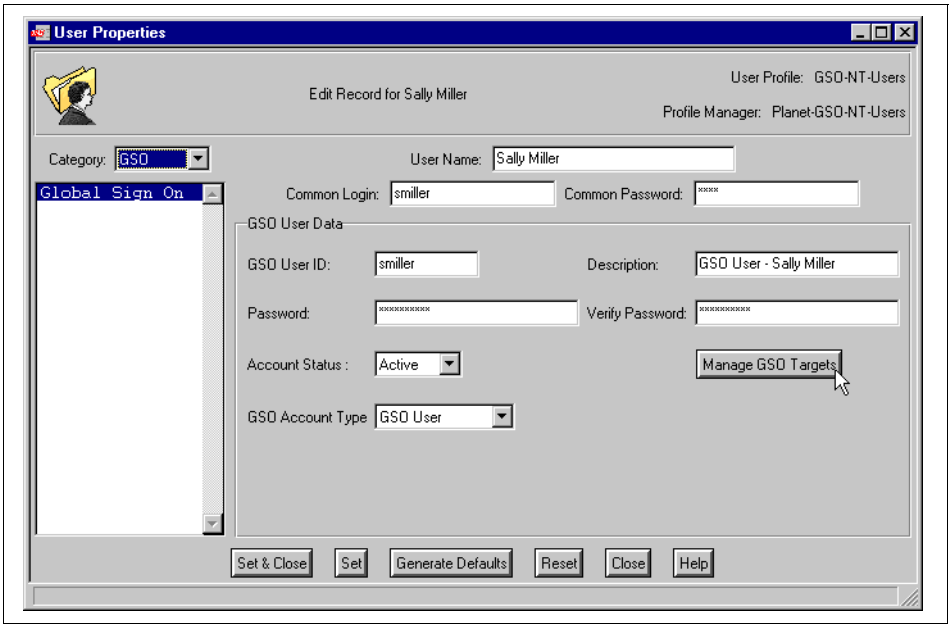


Figure 79. Selecting Manage GSO Targets

4. Click on **Manage GSO Targets,** which opens the Manage GSO Targets window (Figure 80).

Figure 80. Manage GSO Targets

5. Click on **Add** to open the Add Target - Select Target Type window (Figure 81). As you add each target, it will appear in the Defined Targets list. Note that if you inadvertently click **Close** on this window, you will not have actually lost the targets you have added so far. You can recover by reselecting **Manage GSO Targets** from the User Properties window. If, however, the User Properties window gets closed before the **Set** or **Set & Close** option is run, all work will be lost.
6. From the Add Target - Select Target Type window, you must select a target type from the Available Target types list. You must also enter a name for the new target. When you have selected a target type and entered the target name, you should click **Select & Close**. If you click **Close** from this window, the action is similar to cancel or dismiss; you will be returned to the Manage GSO Targets screen.

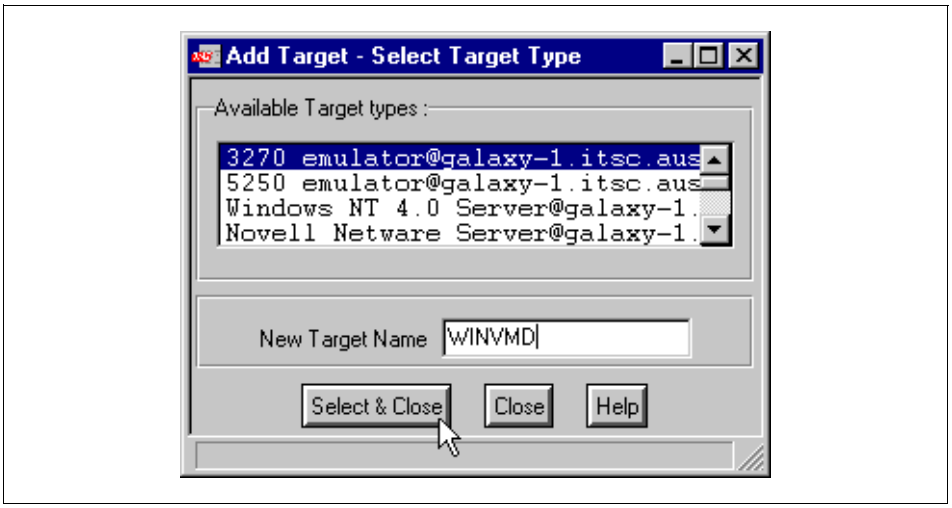


Figure 81. Select Target Type

7. If you do **Select & Close** from the Add Target - Select Target Type window, the Add GSO Target window will appear. This is made up of three distinct areas: Target Information, Target User Information and Target System Information.

   Note: The Add GSO Target window that now appears is rather big, such that details could not be seen if it was shown here in its entirety. For this reason, the figures that follow only show a portion of that window.
8. The Target Information area (Figure 82) shows the target name as you entered it on the Add Target - Select Target Type panel.

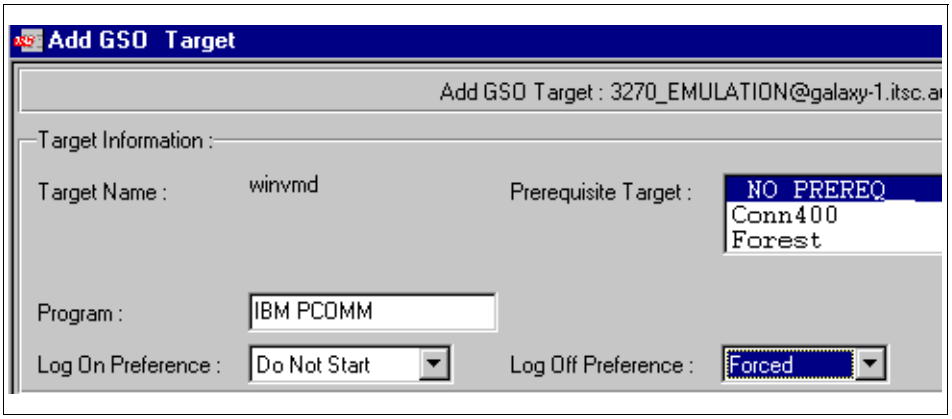


Figure 82. Target Information - Top Part of Add GSO Target Screen

Enter the following:

1. Program – Enter the name of the GSO program object that should be used by the Launcher when logging on to this target. Most types of targets do not require this to be filled in because the Launcher will use the first program that has the same Target_Type value as the target object. The types of target that do require a program name to be entered are the 3270 and 5250 emulators. This name must match the name given when the program is added at the client (see 6.2, “Adding GSO Programs” on page 152). There is no pop-up list; so you need to know the exact names as they are defined in the program database.
2. Log On Preference – This field is used to specify the action GSO is to take for the target after the user signs on to GSO. **Start and Logon** means that GSO will automatically start and log on to the target when the user signs on to GSO. **Do Not Start** means that GSO will not initiate the target. The user will need to select the target and log it on when required. **Start Only** means that the target will be started, but the user will not be logged on. Note that not all targets support all logon preferences. You should check 6.4, “Supplied Targets” on page 171, for the options available. There is no cross-check made by GSO at this point in the definition. If a non-supported option is selected, the target will fail at runtime with an error. The preference can be changed from the GSO Launcher at runtime. It does not necessarily need to be changed from TME 10 GSO User Administration. The ability to change logon preferences from the GSO Launcher can be useful in allowing users to customize the way their targets behave. For example, if the majority of users had a target that they used only occasionally, the setting for the target could be set to **Do Not Start**. The minority of

users who used this target at every session could then change the preference to **Start and Logon**.

3. Log Off Preference – This field is used to specify the action to be taken when the target is logged off. Options can be **Forced**, which is basically logoff immediately, or **Graceful.** The other option that can be set is **Not Allowed.** This option should be used for targets that do not support the logoff interface, or if you want the target to remain logged-on even when the user signs off the Launcher GUI. As with the Log On Preferences, the target will fail at runtime if an unsupported option is selected, but the preference can be changed from the GSO Launcher.

> **Note**
>
> The GSO 2.0 release used for writing this book contained a bug related to setting the preferences. If you are running the TME desktop from a Windows NT or 95 system, and you do not select a logon and logoff preference, the resulting target object will cause the GSO Launcher to crash during target logon or logoff. Therefore, when adding or updating a user's target, always choose a logon and logoff preference. Note that the end users can always change the preferences according to their needs from the GSO client.

4. Prerequisite Target – This option can be used to select a target that has to be started or logged on prior to this target. This can be useful where, for example, a connection has to be acquired before a session. Note that both targets will appear on the user’s target list on the GSO Launcher panel after they sign on to GSO. There is not a mechanism whereby a prerequisite target can be defined and not be visible to the user. The Log Off Preferences for this target and the prerequisite target should ideally have the same value. GSO will attempt to log off the prerequisite target when this target is logged off. Conversely, GSO will not allow a prerequisite target to be logged off before the target that specifies it as a prerequisite. Note that you can only select prerequisite targets from the Prerequisite Target list. This list shows all targets added for the user to date. You need to have added the prerequisite target before you add the target that prerequisites it. It is possible for multiple targets to specify the same prerequisite. For example, if you have an Client Access/400 connection target, you could have more than one 5250 emulation session target that prerequisited the connection.

9. The Target User Information (Figure 83) relates to the information GSO uses to log the user onto the backend system.

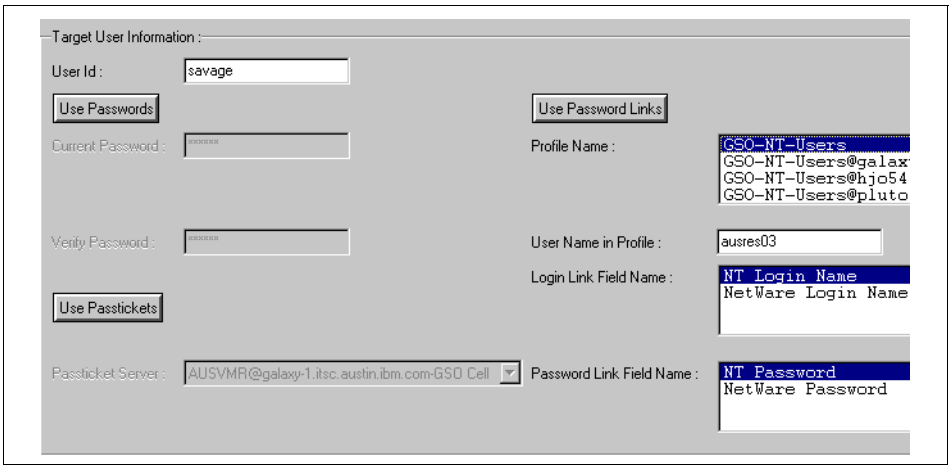


Figure 83. Target User Information - Middle Portion of Add GSO Target Screen

There are three main options:

1. Use Passwords – You need to enter the user ID and password that the user has been set up with on the backend system. The password has to be entered twice for verification.
2. Use Passtickets – You need to enter the user ID by which the user is known on the backend system. You will also need to enter the name of the passticket object maintained by the GSO server. Passtickets are currently supported with 3270 emulation programs only.
3. Use Password Links – This is a mechanism whereby you can have the target password set to the same value as a password in another profile. In this release of GSO, linking is only supported to Windows NT or Novell NetWare profiles. You need to select either **NT** or **NetWare** for the **Login Link Field Name** and the **Password Link Field Name**. You need to select the **Profile Name** of the profile manager or endpoint that contains the user profile. You also need to enter the user ID in the User Name in Profile field. (Note: This is actually the TME user name found in the top-most field on the User Properties window, not the name in the common login field). Password linking can be a useful feature when setting up targets; however, this is at initial setup only. If the user changes the password on a target at runtime, there will be no automatic link-back to change the password for NT or NetWare in the TME 10 GSO User Administration profile(s). It is possible to set up

tasks in conjunction with Tivoli Event Management to have password changes synchronized; you should consult the Tivoli documentation for information. In general, changes in passwords by users at runtime should always be done from GSO and not from the native system. If a user logs on to a system natively, meaning not through GSO, and changes his or her password, the password in the GSO database will be out-of-sync, and future GSO logons for the target will fail.

If you do use password links, then whenever the password in a linked profile is changed in TME 10 GSO User Administration, it will also be changed with respect to the user target entry. Note, however, that the new password will only be passed to GSO if the user profile is distributed to the GSO cell. You need to plan for how linked passwords are to be used and maintained within your environment.

10. Target System Information can vary depending on the target type. The actual fields are determined from the schema file and are used with the program template to provide run-time information specific to this instance of the target type. There can be up to three data entry fields. In the example shown below in Figure 84, which is for IBM PCOMM, the fields that can be entered are Application and System. Fields may be optional or required, depending on the setting coded in the schema file.

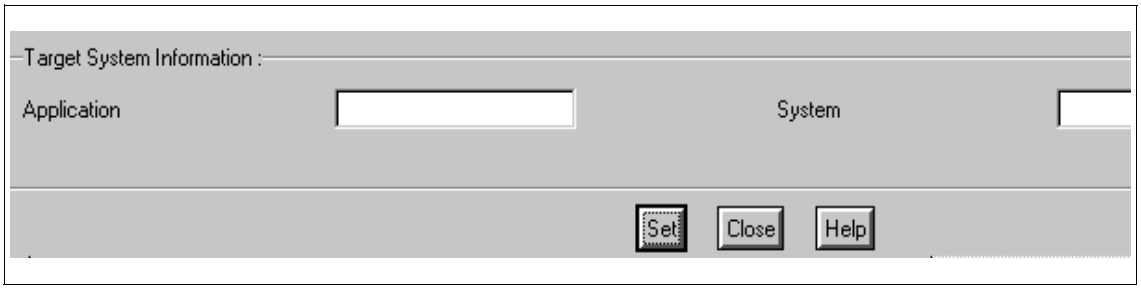


Figure 84. Target System Information - Bottom Part of Add GSO Target Screen

11. When you have entered all the fields need for the target, you should click **Set** to save the settings. Note that the values will not yet have been included as part of the user profile; this is only done when **Set** or **Set & Close** is clicked on the User Properties dialog from which you are managing GSO targets (see Figure 79).

    You can change values on the **Add GSO Target** screen and click **Set** as needed. You will need to **Close** the dialog before you add another target, even of the same type, because each target must have a unique name.

There is no **Set & Close** option for this dialog; if you close without clicking **Set**, the values entered will be discarded.

12. When you have finished managing the targets for a user, you need to **Close** this dialog (Figure 80) to return to the **User Properties** dialog (see step 3 above). You then need to click **Set** or **Set & Close** to save the values to the user profile.

13. When you have finished adding targets for users, you need to distribute the user profiles to the GSO cell. A common way to do this is to highlight at least one profile and then click the **Distribute...** option from the **Profile** pull-down menu (Figure 85). Once a profile is highlighted, all user profiles get distributed, not just the ones highlighted. If you use the distribution options as described in the next step, only changed fields get updated in the GSO cell. If you had a situation where you had changed several profiles but only wanted to update a subset in GSO, then you need to copy the profiles to a temporary user profile for distributIon.

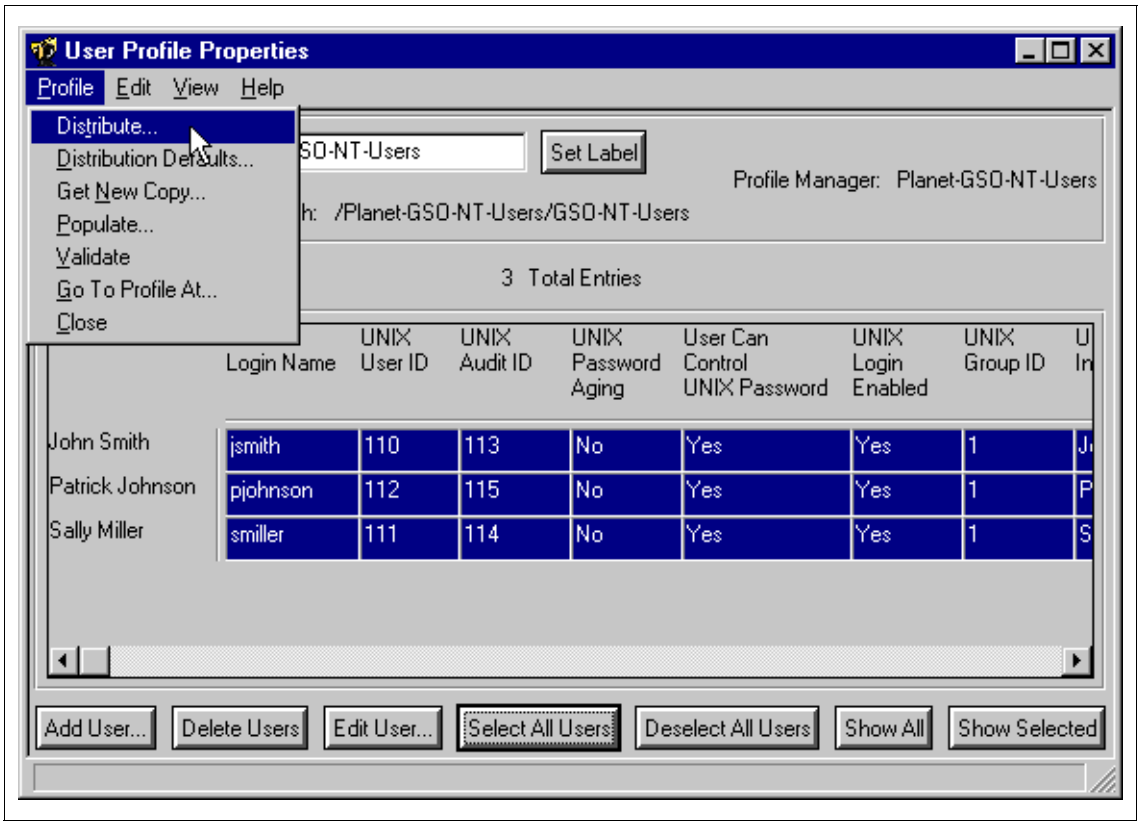


Figure 85. Selecting Distribute User Profiles

14. On the **Distribute Profile** dialog box (Figure 86), you must select the GSO cell as one of the subscribers. You must also select **All levels of subscribers** for the distribution to be successful. You may choose to completely overwrite the data at the subscription endpoint by selecting **Make each subscribers profile an EXACT COPY of this profile**. Normally, you would select **Preserve modifications in subscribers' copies of the profile**. This option will only update the endpoint copy if the data is new or if changed data has the same value at the subscriber as the previous value held in TME. This means that a target password will not get overwritten if a user has changed it in GSO since the last time the data was distributed.

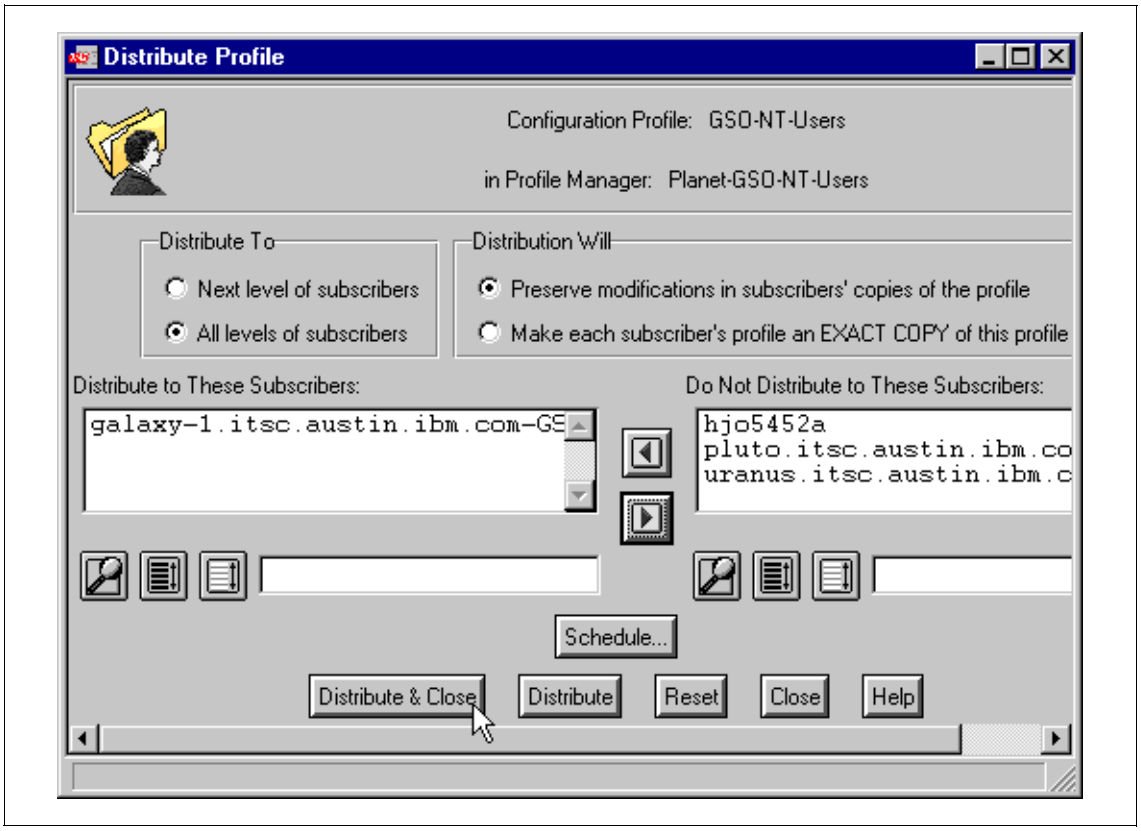


Figure 86. Distribute User Profiles

6.4 Supplied Targets

This section details the specific data that needs to be entered when adding the targets supplied with GSO. Each subsection has the same format and details:

1. The GSO client platforms the target is supported on.
2. Program interfaces supported - These are the interfaces for logon, change password and so forth. You may need to change the program path for these using the **Advanced...** option when you add the target program (see 6.2, “Adding GSO Programs” on page 152, and Figure 71 on page 155).
3. Program Set-Up Values - These are the target-specific fields that form part of the target program definition (see 6.2, “Adding GSO Programs” on page 152). These are explained here, but you may also wish to review the related Program Template File (PTF). These can be found on the GSO clients under the <ibmgso>\template directory.
4. Target System Information - These are the values that need to be completed when adding the target for a user (see 6.3, “Adding Targets for Users” on page 161). Again, you may wish to review the related Program Template File and the schema file since these fields are derived from the schema. The IBM Schema File, ibmgso.sch, is installed as part of the GSO server and can be found under the /var/gso/schema directory on UNIX and under the <ibmgso>\schema directory on NT. This file contains schemas for all the targets supplied with GSO.
5. Notes - General information and notes relating to setting up the target, the runtime environment, and any special considerations that you should know of. The version of target products supported is also given, but you should check the GSO Readme files for the latest information.

6.4.1 Lotus Notes

This section describes the Lotus Notes client as a GSO target.

Client Platforms Supported

The Lotus Notes target is supported on OS/2 Warp, Windows NT and Windows 95 clients.

Program Interfaces Supported

The following table lists the supported program interfaces.

Program Interfaces	Supported
Start	No
Logon	**Yes**
Change Password	No
Logon and Change Password	No
Forced Logoff	No
Graceful Logoff	No

Program Advanced Values

You may need to change the location of the notes.exe file.

Program Set-Up Values

None.

Target System Information

Application — You should enter the name of the notes ID file here. The name does not have to include the .id suffix. Note that the user ID entered when adding the target should be the user's short name.

Workstation — You can, if you wish, enter a value in this field; however, it has little to no effect as logon, and change password actions are only effective on the client machine where the target is being run (see the notes below). If you do enter a value, it should be equal to the TCP/IP hostname of the client machine.

Notes

GSO only supports Lotus Notes 4.0 and above.

If the Notes single logon feature is installed, you must remove it. To do this, issue the command:

```
nslinst -delete
```

GSO uses a Lotus Notes-provided interface to cause it to be called to perform logon. Before you can use GSO for Lotus Notes targets, you will need to change the notes.ini file.

On NT, the default directory for this file is \winnt; on Windows 95, the default is \windows. On both these systems, you should add the line

EXTMGR_ADDINS=ngs0452.dll if you are using Lotus Notes 4.5.2 or higher, or add EXTRMGR_ADDINS=ngs451.dll for lower-level Lotus Notes 4.X systems.

On OS/2 Warp, the default directory for this file is \notes\data. You should add the line EXTMGR_ADDINS=igs0452.dll if you are using Lotus Notes 4.5.2 or higher, or add EXTRMGR_ADDINS=igs451.dll for lower-level Lotus Notes 4.X systems.

If you do not change the notes.ini file, Lotus Notes will continue to prompt the user to enter the password even if the target has been coded. If you uninstall the GSO client at any time, you will need to comment out this line before you can log on to Lotus Notes natively.

GSO provides a logon interface only. The target can be started automatically from GSO or on being selected by the user. A GSO logon for the Lotus Notes target will also be performed if the user runs Lotus Notes natively.

GSO does not provide a change password interface. The user must change their password using the Notes provided facility. However, because of the notes.ini file change, GSO is invoked from Notes when the user elects to change the password. The GSO Change Target Password pop-up is displayed. When the user changes the password, it is changed in GSO and also in the *.id file on the local client machine. Although GSO allows you to enter a workstation name as part of the target system information, it is not actually used at runtime; it's purely a method of identifying which workstation this target was intended for. If the same target is being used on multiple workstations, you will need to ensure the *.id file is copied or updated to the other workstations because the password in their ID files will now be out of sync with GSO. The other option is to code separate targets for each workstation, hence the workstation option.

If you have more than one Lotus Notes login, you will need to code separate targets for each.

Lotus does not provide a logoff interface that GSO can use; you must set the Logoff Preference to **Not Allowed** when you add the target. If the user closes Notes or uses PF5 to logoff, the target status in GSO will still show as logon complete. However, if the user re-invokes Notes, either natively or by using the target logon option from the GSO Launcher, GSO will log the user on again.

6.4.2 Novell NetWare

This section describes the Novell NetWare target support.

Client Platforms Supported

Novell NetWare Server and Novell NetWare NDS targets are support on OS/2 Warp, Windows NT and Windows 95 clients.

Program Interfaces Supported

The following table lists the supported program interfaces.

Program Interface	Supported
Start	No
Logon	**Yes**
Change Password	**Yes**
Logon and Change Password	No
Forced Logoff	**Yes**
Graceful Logoff	No

Program Set-Up Values

Command File Path — This field can be used to specify the full path and name of a command file (.BAT) to be run by GSO after the user is logged on to NetWare. The primary purpose is to issue `NET USE` or map commands for the local drives. The default value for this field is `NULL`.

Target System Information

Server Name — Used for bindery only, this is the name of the Novell Server the user is to be logged onto.

Context Name — Used for NDS only, this defines the location in the NDS tree to be used. Entries should be in the form of a directory, for instance \root.

Notes

GSO supports Novell NetWare Client 4.x on Windows NT/Windows 95 and Version 2.12+ on OS/2 Warp. There are two targets provided, one to support logon to Novell NetWare NDS and the other to log on to the Novell NetWare server (using bindery mode).

IntraNetWare Requester

Novell IntraNetWare Requester (bindery and NDS) is supported on Windows NT and Windows 95. All levels of the product are supported for bindery. NDS logon, however, is not supported with Versions 3.x.

6.4.3 Windows NT 4.0

This section describes Windows NT 4.0 as a target. Note that GSO 2.0 does not support a domain login; it supports logons to map shared resources (as if you entered a `net use x: \\<server>\resource` command).

Client Platforms Supported

Windows NT targets are support on Windows NT and Windows 95 clients. While the target is not provided on OS/2 Warp, it is possible to perform an NT logon from OS/2 Warp by using the **LAN Server Logon to a Domain** target (see 6.4.5, “LAN Server Logon to a Domain” on page 178).

Conversely, it is possible to use this target to log on to a LAN Server Domain from Windows NT (see notes below).

Program Interfaces Supported

The following table lists the supported program interfaces.

Program Interface	Supported
Start	No
Logon	**Yes**
Change Password	**Yes**
Logon and Change Password	No
Forced Logoff	**Yes**
Graceful Logoff	**Yes**

Program Set-Up Values

None.

Target System Information

Local Device The local drive assigned to the shared resource, for example Z: or LPT1. This is an option field; if it is not entered, a

driveless connection will be made. Note that specifying an asterisk (*) as a device identifier is not supported.

Domain — The name of the NT server to be logged onto. For password change operations, this must be the fully qualified name, for example lsmachine.ls.com.

Resource — The name of the resources to be accessed in the form of \\domain\resource, for example \\lsserver\lsapps.

Notes

Domain and Resource are required. If a local device is not specified, the resources will still be accessed, but a local device will not be assigned.

You can use this target from Windows NT to access a LAN Server Domain (LAN Server targets are not available on NT). You will need to specify a shared resource, but you could use the net logon file since this is always shared.

Note that if you have users who will log on to the same LAN from both Windows and OS/2 Warp, you will need to add two targets for them. Unfortunately, because targets are defined at the user level, one or the other target will fail stating `program not found` depending on where they logon from. For example, when the user logs on from Windows NT, the NT-defined target will log him or her onto the LAN. The target defined to be used from OS/2 Warp will fail, however, because the GSO LAN Server program is not available on NT. You may wish to consider developing your own target if you have this situation.

6.4.4 Client Access/400

This section describes the Client Access/400 (CA) as a GSO target.

Client Platforms Supported

Client Access/400 is supported on Windows NT and Windows 95 only.

Program Interfaces Supported

The following table lists the supported program interfaces.

Program Interface	Supported
Start	No
Logon	**Yes**
Change Password	**Yes**

Program Interface	Supported
Logon and Change Password	No
Forced Logoff	No
Graceful Logoff	No

Program Set-Up Values

Command File Path — This field can be used to specify the full path and name of a command file (.BAT) to be run by GSO after the user is logged on to Client Access/400. This file could be used to issue net use commands to map local drives to shared resources. The default value for this field is `NULL`.

Target System Information

System — The name of the system where the user is to be logged on.

Note on System

The system name needs to match the name used by the connections program. These names are fully qualified; so you should enter a fully qualified name here (for example: machine.ls.com).

Notes

GSO supports Client Access/400 Version 3.1.2. The Readme file states that ServicePak SF43837 is required. We found we needed to install ServicePak SF47544 on both the client and the server.

You should do the following to prevent Client Access/400 from prompting for a sign on during Windows startup prior to GSO logging the target on for the user:

- Ensure that the Client Access/400 Login Service Check is not configured to run at startup. You should remove it from the startup folder.
- Make sure that there are no network drives or printers configured to automatically reconnect at startup. You should use a command file with the GSO target to have these resources mapped after GSO has logged the user on.
- On Windows 95 clients, you should disable password caching to allow GSO to handle the passwords. You will need to do this from **Start** -> **Settings** -> **Control Panel**. You should then double-click the **CA added Passwords** icon and deselect password caching.

You should review the GSO *User Administration Guide* for information on mixed environments, meaning those where there are connections that are, and those that are not, configured as GSO targets.

Client Access 5250 emulation cannot be installed on the same machine as the IBM Personal Communication product. The two products are, however, very similar. To code Client Access 5250 emulation as a GSO target, you should use the IBM Personal Communications 5250 program and the 5250 emulator target type. These are described in 6.4.9, “3270 and 5250 Emulation” on page 182.

The primary difference between using Personal Communications 5250 and CA 5250 is that Client Access/400 requires you to have a connection. If you attempt to use CA 5250 before a connection has been started, you will receive the following error in the GSO error log when you add 5250 emulator targets for users that use CA:

```
The Client Access on xxxxx target returned a severe error: rc=6025 (0x1789)
```

Therefore, you should ensure that the IBM Client Access/400 target for the connection is defined as being a prerequisite. See 6.3, “Adding Targets for Users” on page 161.

You will also need to ensure that the Client Access/400 directories containing the pcshll32.dll and addconn.dll (these are normally under the <client access>\emulator and <client access>\shared directories) are included in the Windows PATH environment variable. You can check this from **Start** -> **Settings** -> **Control Panel** -> **System** -> **Environment**.

6.4.5 LAN Server Logon to a Domain

This section describes the LAN Server logon as a GSO target.

Client Platforms Supported

This LAN Server target is supported on OS/2 Warp only.

While the target is not provided on Windows NT, it is possible to perform a LAN Server logon from NT using the Windows NT 4.0 target, as described in 6.4.3, “Windows NT 4.0” on page 175.

Conversely, it is possible to use this target to log on to an NT domain from OS/2 Warp; see the notes below.

Program Interfaces Supported

The following table lists the supported program interfaces.

Program Interface	Supported
Start	**Yes**
Logon	**Yes**
Change Password	**Yes**
Logon and Change Password	No
Forced Logoff	**Yes**
Graceful Logoff	**Yes**

Note Forced Logoff

If you want to use forced logoff, you must change the default path and filename to the path and filename for the LAN Server logoff.exe. By default, this is C:\MUGLIB\LOGOFF.EXE.

If you leave the default unchanged, a graceful, not a forced logoff, will be performed.

Program Set-Up Values

None.

Target System Information

Domain The name of the domain where the user is to be logged on.

Notes

GSO supports OS/2 LAN and Warp Server Clients Version 4.x.

You can use this target to log on to an NT domain from OS/2 Warp. Note that if you have users who will log on to the same NT domain from both OS/2 Warp and Windows, you will need to add two targets for them. Unfortunately, because targets are defined at the user level, one or the other target will fail stating `program not found` depending on where they log on from. For example, when the user logs on from OS/2 Warp, the OS/2-defined target will log him or her onto the domain. The target defined to be used from NT will fail, however, because the GSO NT Server program is not available on OS/2 Warp. You may wish to consider developing your own target if you have this situation.

6.4.6 LAN Server Local Logon

This section describes the LAN Server local logon as a GSO target.

Client Platforms Supported
This LAN Server target is supported on OS/2 Warp only.

Program Interfaces Supported
The following table lists the supported program interfaces.

Program Interface	Supported
Start	**Yes**
Logon	**Yes**
Change Password	**Yes**
Logon and Change Password	No
Forced Logoff	**Yes**
Graceful Logoff	**Yes**

Note Forced Logoff

If you want to use forced logoff, you must change the default path and filename to the path and filename for the LAN Server logoff.exe. By default, this is C:\MUGLIB\LOGOFF.EXE.

If you leave the default unchanged, a graceful, not a forced logoff, will be performed.

Program Set-Up Values
None.

Target System Information
None.

Notes
GSO supports OS/2 LAN and Warp Server Clients Version 4.x.

This target should be used where a logon is required to a PEER machine or to a machine that is outside of a domain. The domain logon target, see 6.4.5, “LAN Server Logon to a Domain” on page 178, should be used for logons within a domain.

6.4.7 LAN Server Manage Passwords in a Domain or on a Server

This section describes the LAN Server manage passwords in a domain or on a server GSO targets.

Client Platforms Supported

The LAN Server manage passwords on a server is supported on OS/2 Warp only.

The LAN Server manage passwords in a domain is supported on OS/2 Warp and Windows 95; it is not supported on Windows NT.

Program Interfaces Supported

The following table lists the supported program interfaces.

Program Interface	Supported
Start	**Yes**
Logon	No
Change Password	**Yes**
Logon and Change Password	No
Forced Logoff	No
Graceful Logoff	No

Program Set-Up Values

None.

Target System Information

One of the choices below depending on the target.

System The name of the system where the password is to be changed.

Domain The name of the LAN Server domain controller.

Notes

GSO supports OS/2 LAN and Warp Server Clients Version 4.x.

Passwords should normally be changed using the LAN Server logon to a domain or LAN Server local logon targets (see previous sections). The manage password targets described herein are for use in changing passwords outside of the local logon environment only.

6.4.8 SnareWorks

This section describes the SnareWorks GSO target.

Client Platforms Supported
SnareWorks is available on Windows NT and Windows 95 only.

Program Interfaces Supported
The following table lists the supported program interfaces.

Program Interface	Supported
Start	No
Logon	**Yes**
Change Password	**Yes**
Logon and Change Password	No
Forced Logoff	**Yes**
Graceful Logoff	No

Program Set-Up Values
None.

Target System Information
Cell The name of the SnareWorks cell.

Notes
SnareWorks as a target is described in more details in a separate section, 6.5, “Implementing SnareWorks” on page 186.

6.4.9 3270 and 5250 Emulation

This section describes the 3270 and 5250 terminal emulator targets.

Client Platforms Supported
GSO supports three separate emulation products: Attachmate EXTRA!, IBM e-Network Personal Communications (PCOMM) and Wall Data RUMBA.

Attachmate EXTRA!, Version 6.2 and above is supported on Windows NT and Windows 95 only and can be used for both 3270 and 5250 emulation.

Wall Data RUMBA, Version 5.1 and above is supported on Windows NT and Windows 95 only and can only be used for 3270 emulation.

IBM e-Network Personal Communications, Version 4.2, is supported on Windows NT, Windows 95 and OS/2 Warp. It can be used for 3270 and 5250 emulation.

Note on PCOMM

IBM e-Network Personal Communications replaces the previous IBM Personal Communications AS400/3270 products. IBM Personal Communications AS400/3270 Version 4.1 is supported with GSO for NT, Windows 95 and OS/2 Warp clients. The Windows NT version requires APAR IC17487. OS/2 Warp requires APAR IC18161. There are no stated prerequisites for Windows 95.

The latest product documentation and Readme files should be checked for up-to-date prerequisite information.

Program Interfaces Supported

The following table lists the supported program interfaces.

Program Interface	Supported
Start	No
Logon	**Yes**
Change Password	**Yes**
Logon and Change Password	**Yes**
Forced Logoff	**Yes ****
Graceful Logoff	**Yes**

**Forced Logoff is not supported by Attachmate EXTRA!

Program Set-Up Values

Session Profile — The full path and file name of the session profile to be used for this target.

Session ID — The letter (in upper-case) used to identify the session.

Script File Path — The full path and file name of the script file to be used with this target, for example c:\ibmgso\script\3270_em\tsosampl.lsf.

Target System Information

System — The name of the host system the user is to be logged on to, fully qualified if needed.

Application — The name of the host application to be logged on to; this field is optional.

Note on Application

Application is an optional field for 3270 emulator targets only. It can be useful where you want to do Logon Applid(xxxxxxxx) type functions.

For 5250 emulator targets, you can start an application, menu and so forth by using a `DATA SEND` command. The sample 5250 script supplied with GSO has a commented example. You will find the script, 5250samp.lsf, under the <ibmgso>\script\5220_em directory on the GSO clients.

Notes

Although there are five separate programs available, see 6.2.1, “Available PTFs by Client Platform” on page 160, there are actually only two target types that can be added for users: 3270 emulator and 5250 emulator. Separate PTFs are supplied for convenience and also to allow for differences between emulation products. The target system information is common to all 3270 and all 5250 targets; therefore there was no requirement to have separate target types.

Before you can use a 3270 or 5250 target with GSO, you must associate the session ID with the session profile. You should start the session you intend to use then from the session window:

- For Wall Data RUMBA, select **Options** -> **API** -> **Configuration** and double-click the session ID you want to use.
- For IBM PCOMM, select **File** -> **API Settings** -> **DDE/EHLLAPI**.
- For Attachmate EXTRA!, select **Options** -> **Global Preferences** -> **Advanced** and double-click the session id you intend to use.

The IBM Personal Communications target can be used for both PCOMM and Client Access/400. If you are using this target for Client Access/400, you must also have a Client Access/400 connection target as a prerequisite (see 6.4.4, “Client Access/400” on page 176).

The GSO Launcher may not recognize a graceful logoff correctly when Telnet sessions are used unless you do the following configuration for PCOMM on the user’s workstation: Click on **Start** -> **Programs** -> **IBM Personal**

Communications -> **Start or Configure Session**. Then, click on the **OK** button and choose the respective session (for example, PC - LAN - TCP/IP Telnet 3270 - S/390). Click on **Configure**, then **Configure Link**. Fill in the correct host name or IP address of the mainframe and click the Advanced button. Then, most importantly, check the **Auto-reconnect** option.

Coding Logon Script Files for 3270 and 5250

There are extensive comments in the sample script files provided with GSO and also in the *User Administration Guide*. The tsosampl.lsf is provided for your reference in Appendix D.1, “Supplied Example Logon Script File: tsosampl.lsf” on page 281, and the information it provides is not repeated here. The best way to code a logon script file is probably to take one of the samples and amend it.

Logon script files do not have to be unique to a target instance. You may well find that you can develop a single script and use it for most host targets.

Logon script files have to be named with a .lsf filename extension. They should not be stored in the <ibmgso>\script directory since this directory gets over-written when the product is updated or reinstalled.

You can use the supplied 3270 or 5250 emulator target to communicate to a UNIX system. No change is required in terms of defining the target, you merely create a VT100 session instead of a 3270 or 5250 session. You then need to develop a script to handle the UNIX logon and so forth. Appendix D.2, “Sample UNIX Logon Script” on page 289, contains a partial script for a logon to AIX. It obviously needs to be developed further to handle change password or other conditions within a working environment.

In terms of coding logon script files in general, you will see from the sample and the documentation that there is no go-to or branch mechanism. You need to take care of the order in which you code “look for” and “query” statements to make sure general conditions are not satisfied before more specific conditions.

There can be multiple start and stop groups within a keyword section, but you cannot code a script that will pass from one keyword to the next. For example, if you code a LOGON section followed by a CHANGEPW section, the script when entered for the LOGON interface will end with the last STOP statement in that section. It will not continue to the CHANGEPW section.

The standard GSO action for `QUERY` commands is to end the script and return a message to the user. For example, `QUERY EXPIRED PASSWORD` will update the GSO log with a message and return code and return a message to the end

user in the Launcher panel. The session will remain open, and the user will have to select the change password option to (in effect) re-enter the script at the CHANGEPW section.

The logon and change password interface is used by GSO if the user selects the change password option for a target and the target is not already logged on. This is the only action that will cause a script to be entered at this point.

6.5 Implementing SnareWorks

As introduced in 2.4.4, “GSO Target Systems” on page 25, SnareWorks, from IntelliSoft Corp., is a security framework based on DCE that adds common security features to legacy applications communicating over TCP/IP without any need for program changes or relinking.

The integration of a GSO client into an existing SnareWorks environment, thus making SnareWorks a GSO target, is relatively easy. Though it is beyond the scope of this book to cover the installation and configuration of a SnareWorks environment, we will cover some prerequisites to simplify the process of installation and configuration of a GSO client to support SnareWorks as a target. In the following sections, the term *SnareWorks cell* describes a SnareWorks environment, including clients, servers, and the necessary DCE cell setup for SnareWorks. Note that this DCE cell is separate from the DCE cell used by GSO.

6.5.1 Pre-Configuration Considerations and Prerequisites

Prior to the installation of a GSO client on a SnareWorks client system, ensure that the following considerations and prerequisites are taken into account:

- SnareWorks supports multiple DCE cells; that is, a user on a SnareWorks client can be connected and logged on to more than one SnareWorks cell at the same time. Because GSO cannot determine to which SnareWorks cell a client is connected to in such a multicell configuration, GSO can only support one DCE cell for any single SnareWorks client.

  However, there can be more than one SnareWorks target be defined for a single user, and this may well work as long as the user assures that he or she uses the correct SnareWorks cell corresponding to a particular target definition. Since GSO cannot verify this through any SnareWorks interfaces and problems are likely to occur if this is not guaranteed, it is not recommended to use this scenario at all.

- SnareWorks does not restrict TCP/IP communication intended for any destination that is outside the SnareWorks cells.
- GSO client configuration for SnareWorks requires that there is a SnareWorks client already installed on the GSO client system.
- A DCE intercell setup has to be created between the GSO cell and the SnareWorks cells to which communication is intended (see Figure 87 on page 189). The intercell relation between the GSO cell and the SnareWorks cell enables GSO to log in to the specific cell in order to do administrative tasks, such as changing user passwords. Though this redbook is not intended as a DCE administration guide, mentioned below are some of the steps for enabling DCE intercell communication for DCE 2.2 on IBM AIX. DCE intercell set up can be done using any of three naming services: X.500, DNS (Domain Name System), or LDAP (Lightweight Directory Access Protocol). The following steps apply to DNS since this is the most commonly used naming service. For more details about intercell set up and related administration tasks, we refer you to your DCE documentation.
  - A DNS entry needs to be created such that the cells can find each other through the DNS naming service and to resolve server addresses. Type the following at the AIX command prompt:

    ```
    # touch /tmp/out.file
    # cdscp show cell /.: as DNS >/tmp/in.file
    # cdscp show clearinghouse /.:/\* >>/tmp/in.file
    # mkreg.dce -input_file /tmp/in.file -named_data_file /tmp/out.file
    ```

    As a final step, the contents of `/tmp/out.file` needs to be appended to the forward resolution file of the DNS name server (by default /etc/named.data) and the DNS daemon (named) needs to be refreshed, for example by issuing the following command:

    ```
    # refresh -s named
    ```

    This configuration, as described above, needs to be done for every cell involved in this cross-cell setup, meaning at least the GSO and the SnareWorks cell. If you have more than one SnareWorks cell, these entries will also have to be added to the DNS configuration for these cells.

Tip

On IBM AIX, the `mkreg.dce` command, the inclusion of the data into the DNS file, and the `refresh` command, as explained above, can be done in a single step using the SMIT tool with the following fastpath:

```
# smit mkdceregister
```

Please read the SMIT help text for further instructions.

  - At least one Global Directory Agent (GDA) needs to be configured in each participating DCE cell. On IBM AIX, for example, this can be done through SMIT:
    ```
    # smit mkgdad
    ```

- After DNS has been set up and a GDA is running in each cell, a trust peer relationship between the participating cells must be established. Run the `registry connect` subcommand of `dcecp`:

  ```
  # dcecp
  dcecp> registry connect <other cell name> \
  -org none \
  -mypwd <my cell_admin password> \
  -fgroup none \
  -forg none \
  -facct cell_admin \
  -facctpwd <other cell_admin password>
  ```

  This creates a principal in each cell that is used for cross-cell communication. Note that running this command requires you to have cell_admin's passwords for both cells. For more details on this command, please read the *DCE Administration Command Reference*.

Figure 87 shows the intercell relationship between the GSO cell and the SnareWorks cell. Both are separate cells, but they do exchange information in regards to users that are defined in both.

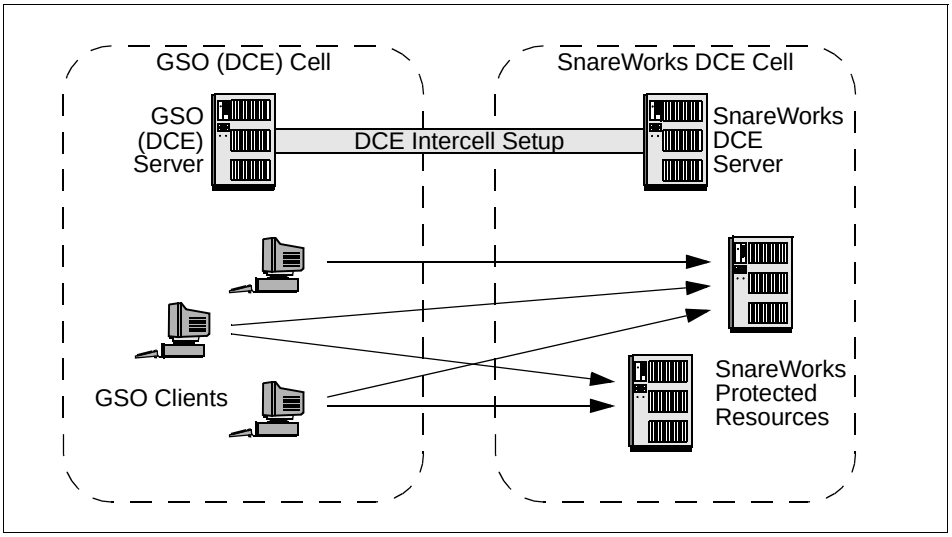


Figure 87. Intercell Relationship between GSO and SnareWorks

6.5.2 Configuration of SnareWorks as a Target

The process of target configuration is a two-step process. Initially, the target has to be created on the GSO server for each user that is going to use the target. Then the configuration of the GSO client needs to be done, enabling the user to use the target from his/her desktop. The two steps are explained below in detail.

6.5.2.1 SnareWorks Configuration as a Target

SnareWorks can be added as a target for an existing user, or can be added as a target while adding a new user. There is only one change in the procedure for adding SnareWorks as a target to an existing user.

Follow the process for editing user properties as explained in Chapter 8, “Managing User Accounts” on page 225 (or in the *User Administration Guide*, shipped with the product) and then follow the steps mentioned below.

To create SnareWorks as a target for an existing user, do the following:

1. Click on the **Manage GSO Targets** button on the User Properties dialog (Figure 88).

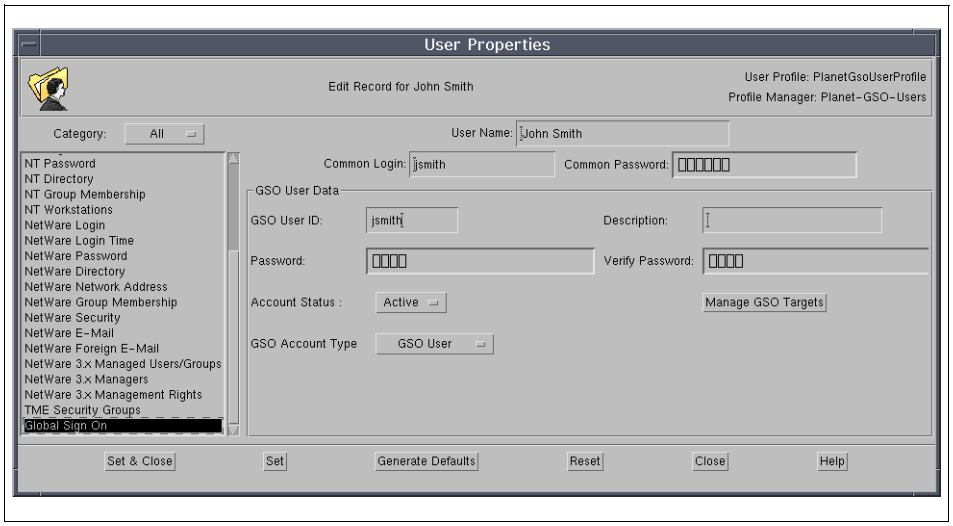


Figure 88. GSO User Properties Dialog

This brings up the Manage GSO Targets dialog.

2. Click on the **Add** button to add a target to this particular user. The Add Target - Select Target Type dialog comes up (Figure 89).

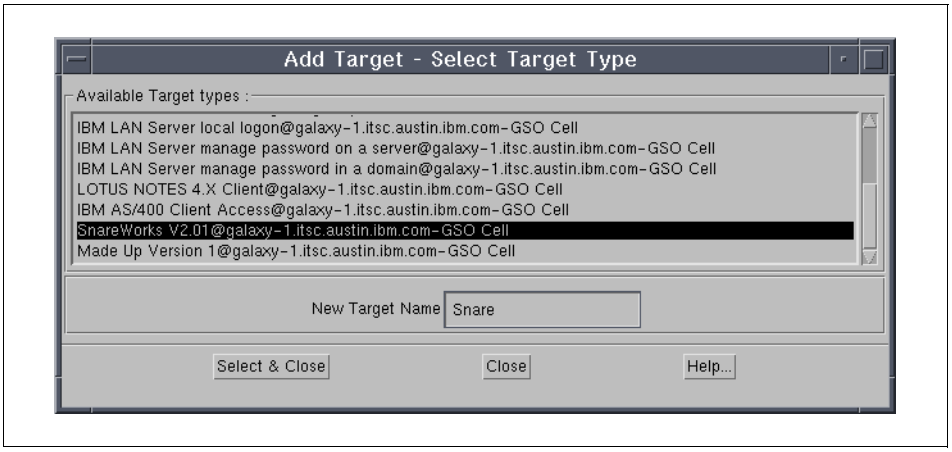


Figure 89. Target Selection Dialog

3. Select **SnareWorks V2.01@<gsocellname>** from the scroll list of available target types, where in our example, gsocellname is galaxy-1.itsc.austin.ibm.com.

4. Type a target name in the **New Target Name** field. This name will be displayed as a target for this particular user on the GSO client desktop.
5. Click on **Select & Close**. This will launch the Add GSO Target dialog (Figure 90).
6. Leave the **Program** field blank.
7. In the Log On Preference selection list, choose the **Start and Logon** option. Unlike with other targets, this option does not mean that the client will log on to the SnareWorks at the time of logging on to GSO because SnareWorks only logs a user on when he/she is attempting to access a protected resource. With the Start and Logon option set, GSO logs the user on to SnareWorks automatically when this user first attempts to access a resource protected by SnareWorks after logging on to GSO and thereby, the SnareWorks target. The default option here is **Do Not Start**. There is another option available: **Start Only**. This option should not be chosen when SnareWorks has been chosen as a target.

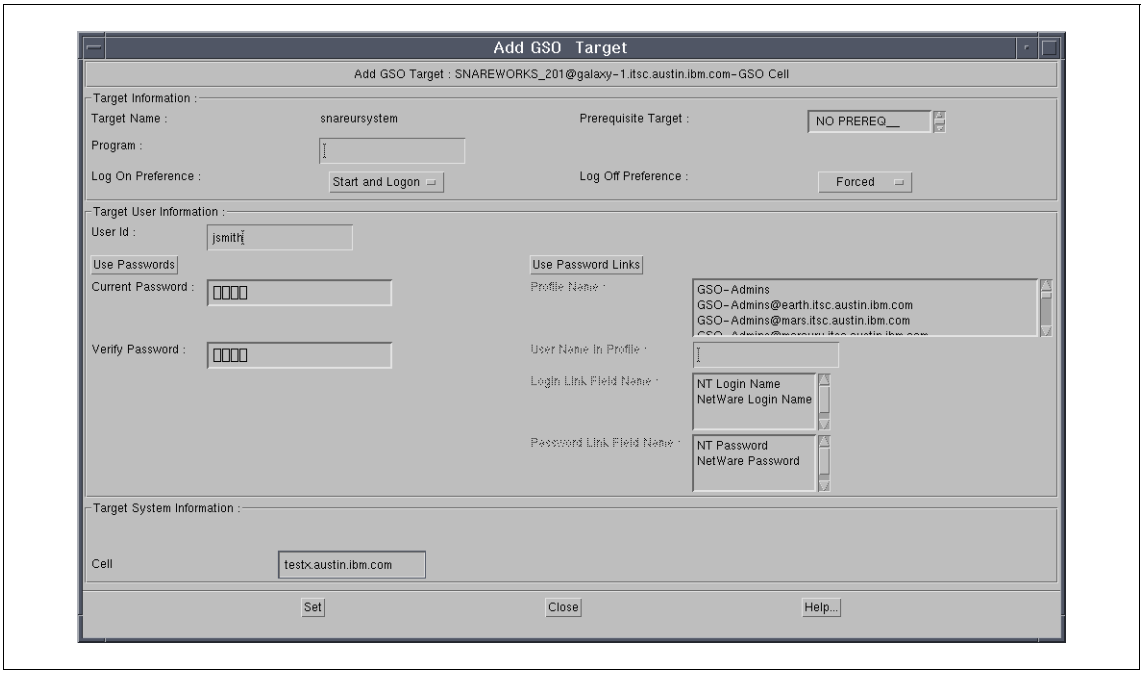


Figure 90. Add Target Dialog for GSO

8. In the Log Off Preference selection list, choose the option that fits your needs. **Forced** is the default option, and this means that the user will be logged off from SnareWorks when he/she logs out of GSO. **Not Allowed**

as an option does not log the user off when he/she quits GSO, and he or she can continue to work with SnareWorks. There is another option available: **Graceful**. This option should not be chosen when SnareWorks has been chosen as a target.

9. Enter the SnareWorks user ID and password in the respective fields (note that the password has to be entered twice for verification).
10. Enter the name of the SnareWorks DCE cell in the Cell field of the Target System Information section.
11. Click on **Set** and then **Close** to save this information for that particular user and to terminate this dialog.
12. Click **Set & Close** again to quit the User Properties dialog. (Do not click on **Close** because the target(s) that have just been added will not be added to the database as is done when **Set & Close** is chosen.)
13. As a last step, the user profiles need to be distributed using the Tivoli Distribute Profile function (see Chapter 8, “Managing User Accounts” on page 225, for more details on this). Distributing the user profiles propagates the profile (and its contents) to the GSO database.

6.5.2.2 Configuration of the GSO Client for SnareWorks

The GSO program for SnareWorks should initially be added to the GSO administration window on the client system. Mentioned below are the steps to add the GSO program to the GSO client database on the client system and also the procedure for logging on to SnareWorks using TCP/IP applications:

1. Double-click the **Global Sign-On Administration** icon on the GSO folder or click on **Start -> Programs -> IBM Global Sign-On Client V2.0-> Global Sign-On Administration**. This will bring up the IBM GSO - Sign-On window.
2. Enter the user ID you wish to log in as and the corresponding password; then click on the **OK** button to log on to the GSO administration task.
3. If you are already logged on to the Global Sign-On Launcher, you will get a window which asks you whether or not you wish to login to the GSO administration task as the same user. If you choose yes, you will be logged into the GSO administration task; otherwise you will be prompted for a new user ID and password.
4. Click the **View** menu option from the administration window.
5. Click on **Programs** to display the programs that are already configured.
6. Now click on the menu option **Programs** and then click on **Add**. This will pop up the GSO - Add Programs dialog (Figure 91).

7. In the Location: field, the directory shown is the default directory (C:\IBMGSO\TEMPLATE). In case you have installed the GSO client in a different directory, specify the directory where the templates exist.
8. Click on the **Get List** button to display or refresh the list of templates underneath.
9. Scroll down the templates list and click on **SnareWorks V2.0** as the template. The Name: and Location: fields are then automatically filled in. If you want to call the program with a different name, you can overwrite the name. In case you have installed GSO in a directory other than the default, then type in the directory including the drive name where the GSO client is installed.

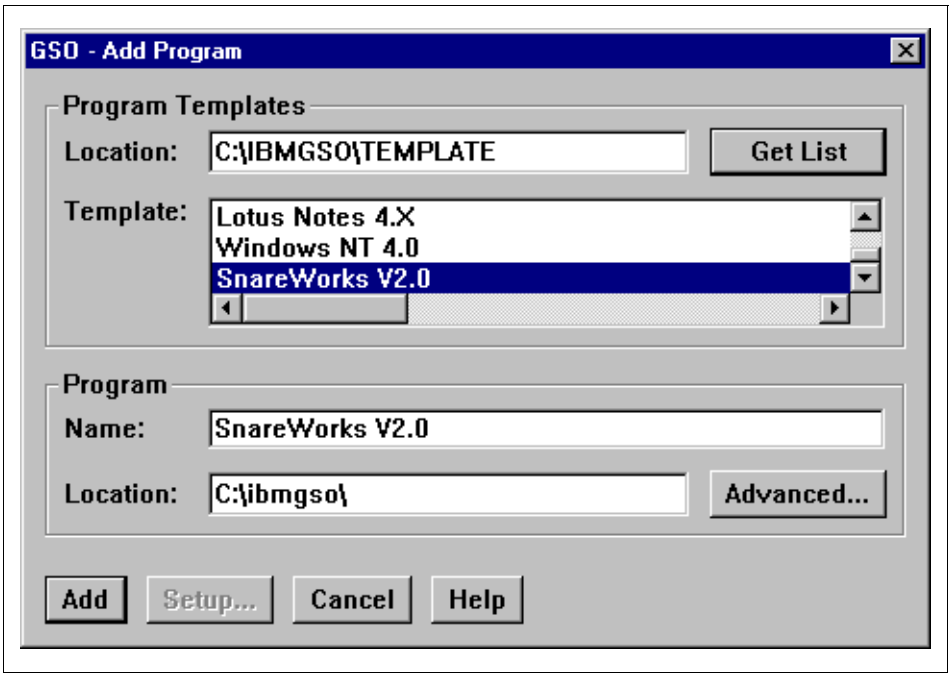


Figure 91. Addition of the SnareWorks Program to the GSO Client Database

10. Click on the **Add** button to add SnareWorks as a program on your client database. The **Advanced** button allows you to change the locations of the GSO executables related to SnareWorks and the settings of their preferences. There is normally no change required in this section.
11. It is recommended to quit the GSO client and the GSO administration task once this addition is done.

After these changes are done, the next time you enter the Global Sign-On Launcher, SnareWorks will be listed as a target on your Global Sign-On Launcher desktop (Figure 92). The name displayed on the system will be the name given to the target in the GSO server while configuring SnareWorks as a target.

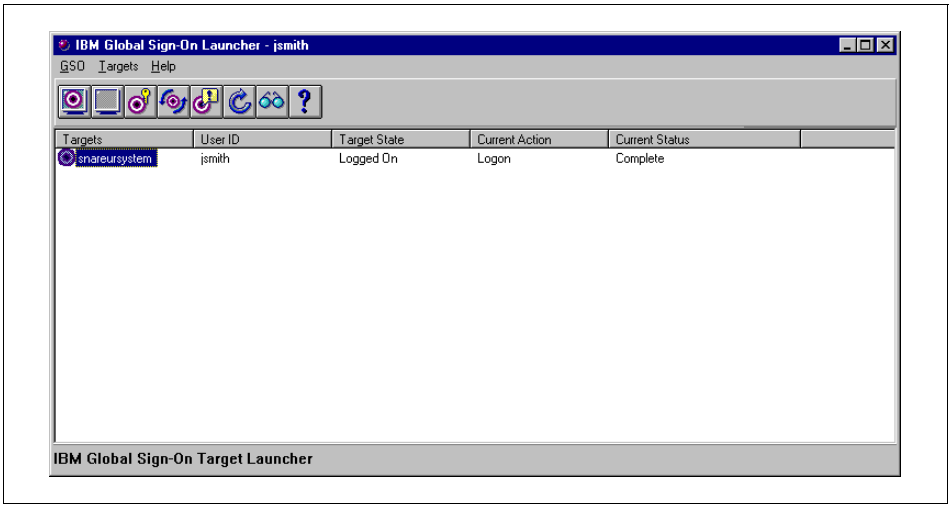


Figure 92. IBM Global Sign-On Launcher

This procedure has added the user into the GSO user database. Every time the user executes a TCP/IP application to connect to the SnareWorks cell, which is also linked to GSO, GSO will pass the user ID and password to SnareWorks. The user may, however, still be prompted for a user ID and a password of the TCP/IP application executed. For example, if the user runs a Telnet application, he or she might still be required to enter a user ID and a password for that particular session, although GSO has taken care of the authentication to SnareWorks.

6.5.3 Changing Passwords

To change the password of a SnareWorks target, you (as a GSO user) must start the Global Sign-On Launcher on the desktop and sign on to GSO with your GSO user ID and password (if not already done). After successful authentication, the Global Sign-On Launcher desktop is displayed with a list of targets defined for this user ID. On this launcher, select (highlight) the SnareWorks target and click on **Targets** and choose the option **Change Password**.

You will be prompted to enter your GSO password as a matter of additional security to ensure that you are the right person requesting such a change. After successful verification, you will be asked to enter the new password for the SnareWorks cell twice (for verification). Once you click on the **OK** button, the password will be changed in the SnareWorks cell as well as in GSO.

This is exactly the same procedure a user would perform for any other target password change. What is special about it is how it works behind the scene. GSO uses the DCE intercell setup to propagate the new password to the SnareWorks cell. This is accomplished by the GSO client, which communicates to the DCE Security server in the SnareWorks cell.

6.6 Using Generic Target Groups

IBM Global Sign-On for Multiplatforms, Version 2.0 allows you to define generic target groups and then distribute the group of targets to users. This can be useful in that it saves having to define each target for each individual user. The limitation is that the user ID used for the target will be set to the GSO user ID (or the common user ID if there is no GSO user ID). If the user has a different user ID on the backend system, a generic target cannot be used.

Generic targets can be set to use passwords, password links or passtickets in the same way as individual targets. If passwords are to be used, however, the password cannot be entered in the generic target. The user will need to enter the password from the GSO Launcher panel before the target is logged on for the first time. For this reason, the Logon Preference for generic targets that use passwords should be set to **Do Not Start** (see 6.3, “Adding Targets for Users” on page 161, and Figure 82 on page 165).

To enter the password from the GSO Launcher, the user should select (click on) the target and then select the change password option. The user will be prompted for his or her GSO password. When that password has been verified, the change password screen will be displayed as shown in Figure 93. The normal action when entering a generic target password for the first time is to select the **Update GSO Database Only** option.

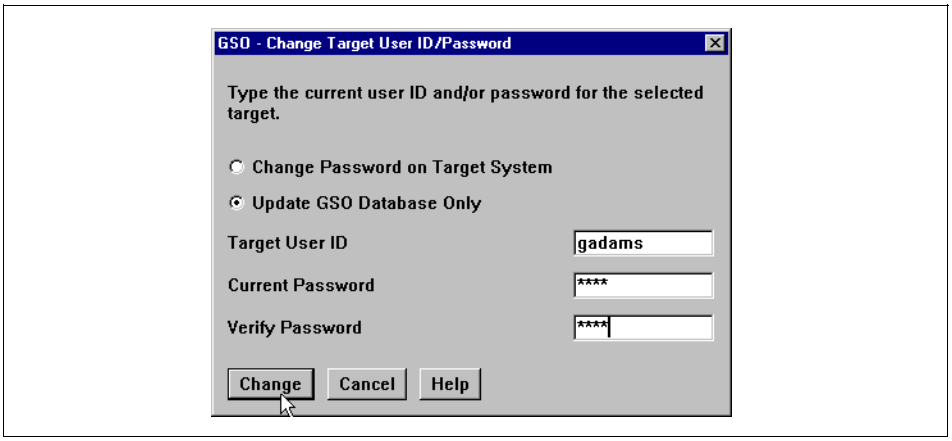


Figure 93. Enter Generic Target Password

The following steps are a guide to creating and managing generic target groups:

1. You will need to select **Target Group Management** from the GSO cell pull-down menu (right-click on the desired GSO cell icon in the appropriate policy region, Figure 94).

Figure 94. GSO Cell Pull-Down Menu

2. You will then get the GSO Cell Target Group Management screen (Figure 95), and you should click on **Add** to add a new target group.

Figure 95. Manage Target Groups

3. The next panel that appears will be the Add a Target Group (Figure 96). This dialog might be confusing because the legend says New Target Group Type rather than New Target Group Name, and you should enter the name you want to give this group of targets.

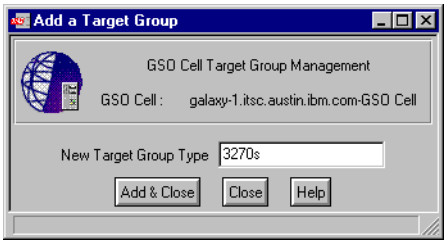


Figure 96. Add a Target Group Name

4. After clicking on **Add & Close** (Figure 96), the Edit Cell Target Group window opens (Figure 97). This dialog has two parts: the Generic Targets part and the Users part.

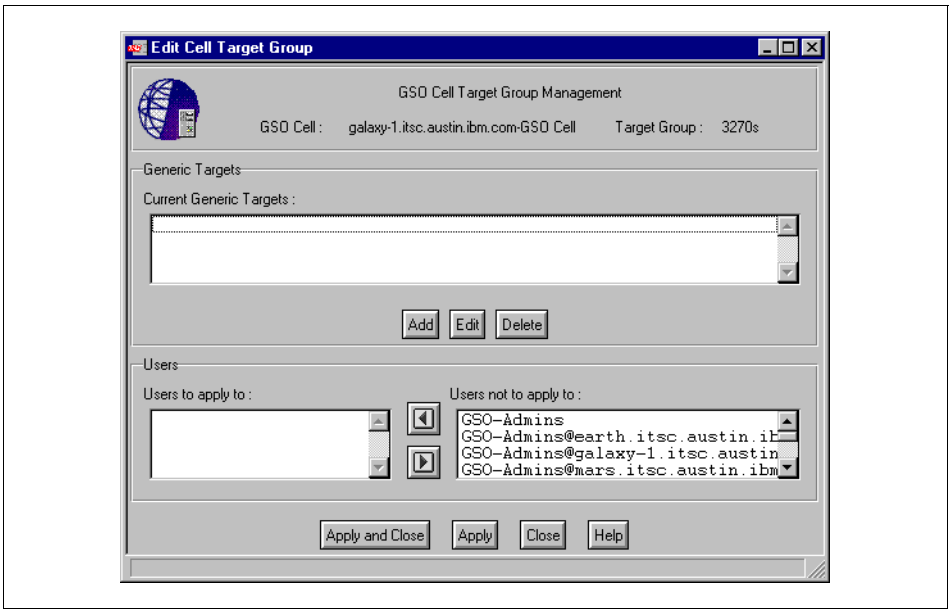


Figure 97. Edit Cell Target Group Dialog

5. Clicking on **Add** in the Generic Targets sections opens the Add a Generic Target dialog, which is very similar to the Add Target dialog shown in Figure 81 on page 164, except that it refers to target groups rather than single targets. You need to select the type of target you want to add to the group and enter the New Target Group Name. You should endeavor to ensure that this is the same name as you used on the Add a Target Group screen above (Figure 96). If you make a mistake, there is no complaint at this point, problems will only arise later. Click on **Select & Close** to proceed.
6. The next dialog window that appears is almost identical to the screen you use to add individual targets (see 6.3, “Adding Targets for Users” on page 161, and Figure 82 on page 165 through Figure 84 on page 168 on how to complete the Target, Target User and Target System information). The main difference is that you will not be able to enter user ID and password as discussed in the opening to this section. When you have entered the target information, you need to click on **Set** and then on **Close**. Back on the Edit Cell Target Group screen (Figure 97), you will notice that the generic target has been added to the list of Current Generic Targets. You can elect to add, edit or delete the targets. Groups can contain targets of all one type or a mixture of types.

7. When you have finished editing the group, you will need to apply the group to a user profile. The bottom half of the Edit Cell Target Group screen shows a list of all the user profiles you can select from. Generic targets have to be applied to a user profiles; they cannot be applied to individual users.
8. When you have completed the generic target group, you will need to distribute it to the GSO cell so that the targets are available for the GSO users. You need to perform the distribution in the same way as you do for any other target, meaning you will need to perform the distribution from a User Properties dialog. See 6.3, “Adding Targets for Users” on page 161, for more details.

This concludes the creation of generic target groups.

6.7 Adding New Targets to the GSO Framework

The GSO product provides a framework to add new target applications. That framework consists of:

- Program Template Files for adding program objects
- Schema files for adding target objects
- The GSO client target launch process for log on, log off, and change password

6.7.1 Techniques for Launching Target Applications from GSO

The GSO client launch process invokes client applications using several techniques, which are:

- By linking with a DLL (Dynamic Link Library) provided by the client application and calling the appropriate client APIs to start the application and pass the user ID, password, server name, and so forth. The GSO product provides a set of wrapper DLLs for launching common client applications, such as NetWare client, NT shared resource mapping, and EHLLAPI for 3270/5250 emulators.
- By calling the client application executable and supplying the user ID, password, server name, and so on as parameters to the executable. The GSO product starts the Lotus Notes and IBM PCOMM applications this way (but does not complete the logon step through the executable). Though not provided with the GSO product, the cc:Mail application, for example, can be started and logged onto by running the following command:

```
Wmail32.exe /Login <mailboxpath> <login name> <password>
```

- By starting the application and then passing the user ID, password, and so forth by directly interfacing with the Windows dialogs. The GSO product interfaces with the SnareWorks product dialogs this way. Many custom applications that only provide a logon through the Windows dialog (and not by API or CLI) can be handled by GSO in this way. This requires a wrapper DLL be written for each Windows-based application as well as a Program Template File and schema file. The IBM GSO Service team has produced several of these wrapper DLLs on special customer requests.
- Some client applications call GSO to get the user ID and password. Lotus Notes, for example, will call a GSO DLL (ngso452.dll) whenever it requires the user's password, instead of displaying the password prompt to the user.
- Some applications use the GSO (DCE) credentials obtained when the user signs on to GSO. The GSO database client works this way.

6.7.2 Adding New Targets

To add a new target to the framework, you will need to analyze the application and follow these steps:

1. How does this application get invoked from the client, and how does the caller pass the user ID, password, and other info?
   - As a command line executable, with parameters?
   - Through an API?
   - Through a Windows dialog only?

   If an API, then it may be desirable to write a DLL that is a library of the target applications' interfaces. This DLL should have entry points for logon, logoff, change password, logon and change password, and so on, that will be invoked by the GSO Launch process. The GSO product has provided such wrapper DLLs for each of the 3270 emulation programs (for example IBM Personal Communications).
2. What parameters get passed to the CLI/API/Window during logon, logoff, and change password operations?
   - Does it require a host name, or domain name, or anything else?
3. Is there an existing target type that has the same set of parameters and behavior?
   - If so, then you do not need to create a new schema file entry.

- If not, then you will have to create a new schema entry for your new target type.

This new entry will be used by the TME User Administration GUI to determine the information that needs to be gathered from the user for this target type (for example: $U = user ID, $D = domain name, $A = resource name).

This new schema entry should be stored in a new *.sch file on all GSO servers. Do not append to the existing ibmgso.sch file. Run Target Type Management to populate the TMR database with the new schema data.

4. Create a PTF (file) for the new target application. Use the template.ptf file, or a similar PTF, as a guide. See, for example, Appendix B, “Program Template Files” on page 255.

   Also, refer to the *IBM Global Sign-On for Multiplatforms, Version 2.0 Programmers Guide* (available online with the product).

5. Use the GSO Client GUI or CLI to create a program using that new PTF.
6. Use the TME User Administration GUI to create a target using that new program.
7. Sign on to the Launcher GUI (or use the CLI) to log on to that new Target.

6.7.3 Program Template Files

Every GSO target must have a program template and a schema. This section does not detail every aspect of a PTF because fairly comprehensive notes are provided in the general blank template.ptf provided with the product. This PTF is printed for your reference in Appendix B, “Program Template Files” on page 255. An example template is also included.

In the MAIN section of the PTF, you specify the target type and the default program name. The default program name is the name that will appear under Template when you are adding programs (see Figure 71 on page 155). You should not file your own PTFs under the <ibmgso>\template directory because they may be overwritten if the product is reinstalled or updated. You should overtype the Program Templates Location field on the Add Program screen with the path for your templates and then click **Get List** to update the Template panel with a list of the PTFs available under this path. The Program Name on the Add Program screen is also derived from the default program name. The Program Location is set to the Directory value you specify in the Settings section of the PTF.

You code a section in the PTF for every logon interface supported, for example Logon, Logon and Change Password, and so forth. These are the

interfaces that will appear in the Advanced Program Path Configuration screen when you are adding programs (see Figure 72 on page 156). The Time-out and Retries values on the Advance Program Path Configuration screen are taken from the values you provide in the Settings section of the PTF. The GSO documentation states that the Min and Max values should be cumulative where you have targets of the same type, for instance multiple 3270s. The PTF is best coded with the default values you expect to use. Runtime adjustments are then made using the Advanced Program Path Configuration screen.

In each interface section, you specify the type of interface program, 32-bit API or CLI, under the capability keyword. You specify the program path and variables to be used under the interface keyword.

There are two types of substitution variables: reserved and unreserved. Unreserved variables, $1 through $7, can be used for program-specific information that will be entered through the Program Setup screen when adding the program (see Figure 73 on page 156). This example shows the setup values for a 3270 emulation target. The related PTF is shown in Appendix B, “Program Template Files” on page 255. You will see that the values shown on the screen are those defined for $1, $2 and $3.

Reserved substitution variables are primarily used for target-specific data that will be entered when the target is added for the user (see Figure 84 on page 168). The two variables that are not related to target-specific data entry are $M, which is used to specify a message string to be output to the GSO log and also on the GSO Launcher panel, and $N, which is used when a user inputs a new password.

6.7.4 Schema Files

Reserved substitution variables to be used for a target are defined through the schema file. There must be a schema file for every PTF. The schema and PTF are associated through the target_type keyword, which must match in both definitions. The minimum that can be specified in a schema is the target_type, target_description, U= ($U is a reserved substitution variable used for user ID), and the type of password (STORED for passwords held in GSO and PASSTICKET where a passticket will be generated).

There can be up to three other reserved variables for a target. These are always defined as $A, $H and/or $D. These variables can be defined as being optional or required. There is a predefined choice as to the label that can be associated with these variables. The choice is limited to the following:

- APPLICATION

- CELL
- DATABASE
- DEVICE
- DOMAIN
- HOST
- PEER
- RESOURCE
- SERVER
- SESSION
- SYSTEM
- WORKSTATION

When coding a schema, you need to choose the keyword that most nearly describes the actual data that needs to be entered. For example, you might use RESOURCE where you wanted the name of a CICS region to be entered. You are not able to label the REGION field. This gives an error at the time you added the schema.

Schemas are unique to target type. You can have multiple PTF files for the same target type. As described in 6.4.9, “3270 and 5250 Emulation” on page 182, GSO supplies a number of different PTFs to allow for the fact that there could be any one of three different emulation programs used, with different program interfaces supported and so forth. For the target-specific data entries needed when adding the target for the users, however, there are actually only two target_type definitions needed, 5250 and 3270, and therefore, only two schemas.

If you are adding a new target application and there is a supplied or existing target_type/schema available, you only need to add a PTF file. The PTF file can be used from the Add Program GSO Administration dialog on the client by specifying its location as described above.

If you have also had to define a new target type, you will also have had to define a new schema. Schema files are held on the GSO server under the /var/gso/schema directory on UNIX and under the <ibmgso>\schema directory on NT. Before targets of the new type can be added for users under TME, the data must be updated to the TMR server. This is done through the **Target Type Management** option from the GSO cell pull-down menu (see Figure 94 on page 196). After you select this option, you will get a screen showing that the target types were populated successfully. They will then be available to add for users.

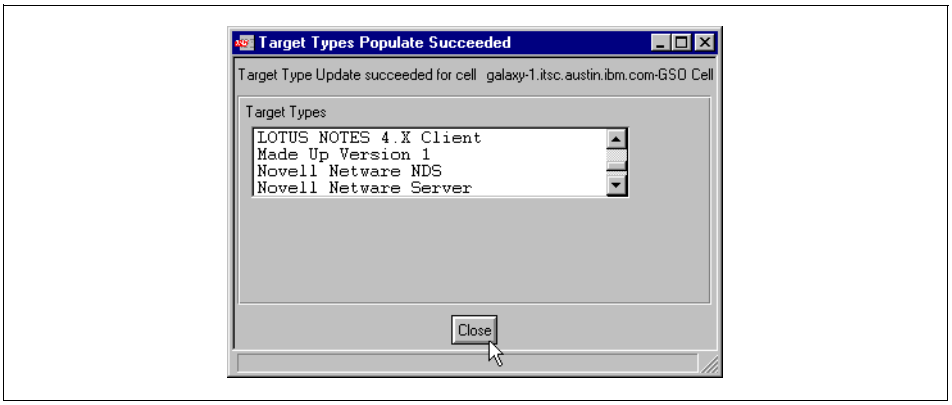


Figure 98. Populate Target Types

The schemas provided for targets supplied with GSO are printed for reference in Appendix C.1, “Supplied Schema File: ibmgso.sch” on page 273. Also included is an example schema file that corresponds to the example PTF provided in Appendix B, “Program Template Files” on page 255.

Adding a program using the program template provides interface entries on the Advanced Program Path Configuration screen for Logon, Logon and Change Password and Logoff Graceful. Adding a target using the schema allows Database, Host and Resource to be entered.

The examples are purely to show what might be coded, rather than any practical or workable target definition. The return code entries are set as using those provided with GSO. The return code keywords given in the sample PTFs result in a standard GSO output message and target status. It is not possible to change the $M related to a return code in a PTF. You can change the return code ranges associated with each return code keyword. To change the keywords themselves and the associated messages requires wrapper code.

Chapter 7. Managing GSO

After GSO is installed, configured and exploited in a production environment, day-to-day management tasks and monitoring activities become important.

This chapter describes methods how the GSO environment can be managed and monitored.

7.1 GSO Management Tasks

The GSO Plus module for Tivoli contains a task library with a number of ready-to-use tasks that support an administrator in the daily management duties. Figure 99 shows the task library with the tasks that it contains.

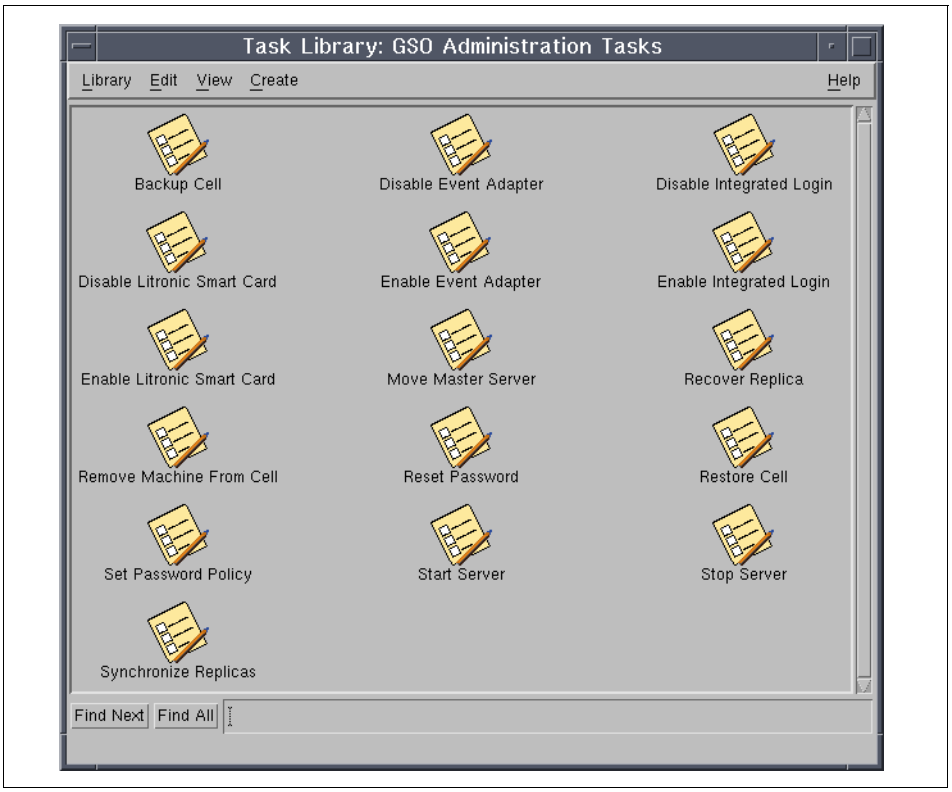


Figure 99. GSO Management Tasks Library

Execution of any of these tasks involves the same basic steps. Depending on a specific installation and configuration, however, task libraries and tasks can

be located and arranged differently to what is shown here. The explanations given here are valid for a basic installation without rearranging the tasks and task libraries.

A task can be thought of a program that executes on specified machines in a specified environment and that carries out a specific function. Running a task can be done in various ways, including:

Manually	An administrator (with the appropriate authorizations) can run the task manually—that is, on request.
As part of a job	A job specifies a task and certain predefined execution parameters and allows the task to be run without specifying them every time.
Scheduled	A scheduled task (or job) can be run automatically at a given time or at given time intervals.

Using the manual method, two common ways to run a task using the Tivoli desktop GUI are: double-clicking on the task's icon or dragging the task's icon over a system's icon in a profile manager. Please refer to the appropriate Tivoli documentation for more information about tasks, jobs and how they can be controlled.

Any tasks could theoretically be run on a single or multiple systems at the same time. Many tasks, however, are specific to a single machine, such as the GSO master server, and running it on any other system would cause it to fail. It is the administrator's responsibility to correctly set up and run the tasks. Jobs could be created that are tailored to a specific environment.

Note on Task Execution Timeouts

By default, tasks have an execution timeout of 60 seconds. If a task does not complete within that period of time, execution might be aborted causing the task to fail. If you anticipate longer execution time, you should increase the timeout value accordingly, or set it to 0 (zero) for no timeout. This can be done on the Execute Task window when a task is being executed.

The following is a brief description of the tasks that are included with the GSO Plus module (refer to Figure 99). Most tasks only require an administrator to specify the system(s) where the task is to be run. Only when necessary, a pop-up dialog window requests additional information from the administrator as described in the descriptions below.

Enable and Disable Integrated Login – These enable or disable the integrated login on any Windows NT or Windows 95 system. No further

parameters are required. This task basically runs the `cfgclient` command on the target system(s) with the appropriate parameters to enable or disable integrated login. When integrated login is enabled, GSO does not require a separate login, but uses the user ID and password from the primary operating system login instead. If integrated login is disabled, GSO prompts the user for a user ID and password when launched.

Enable and Disable Litronic Smart Card – This enables or disables the use of the Litronic Smart Card (see also 5.4.3, “GSO Client System Setup for Smartcards” on page 135) on the specified system(s). No further parameters are required. As with the previous task, this task runs the `cfgclient` command on the specified system(s) to enable or disable Smartcard support. When enabled, GSO requires the user to use his or her Smartcard and the assigned PIN (personal identification number) in lieu of a user ID and a password.

Backup and Restore Cell – This task allows an administrator to create a backup file of all the relevant information (that is the security registry) stored on the GSO master server. A filename for the backup file must be provided as an execution parameter. The resulting backup file resides on the filesystem of the GSO master server. It can be included in the backup and archive procedures for regular files, for example by using ADSM (Adstar Distributed Storage Management). Make sure that you create backups of the GSO master server on a regular basis and when major changes are done to the registry (for example, when a large number of users are to be added). It is a good practice to run this command scheduled at regular intervals. The restore operation allows an administrator to restore the security registry from such a file created with the backup cell task. The created backup file should only be used for a restore operation on the same system. **Note:** Although the registry contains any user passwords and other sensitive information in an encrypted form, care should be taken as far as the location and handling of this backup file is concerned.

Start and Stop Server – This tasks allows for easy stopping and starting of a GSO master or replica server. No parameters are required. Note that this operation can also be done locally on the server(s) with the `gsocfg -start` or the `gsouncfg -stop` commands, respectively (or, on IBM AIX, using SMIT with the fastpath `gso`)

Enable and Disable Event Adapter – This task starts or stops the event adapter on selected systems. No parameter is required for task execution.

Remove Machine From Cell – This task removes a machine’s references from the GSO cell’s database. The task must be run on the GSO master server. Note that this task does not change the configuration on the removed

machine itself; it only removes the references to that machine from the GSO database. This task is typically run when a replica server becomes unavailable for an extended period of time to remove it from the master's list. This prevents the GSO master server from repetitive attempts to update a GSO replica server that is offline. The hostname or IP address of the machine to be removed is required as a parameter.

Move Master Server – This task must be run on an existing GSO replica server, and it converts this replica server into a master server. The existing GSO master server becomes a GSO replica server upon running this task. The cell administrator's password is required as a parameter when running this task. As an option, all other GSO servers can be removed if they are not available at the time of task execution. If the current GSO master server is unavailable when running this task, for example due to a system failure that causes this operation to be necessary, the task will still succeed, but the unavailable GSO master server cannot be reconfigured to a GSO replica server. **Caution:** Unpredictable results can occur if the old GSO master server is brought online after a new GSO master server has been elected.

Synchronize Replica – This task runs on GSO replica server(s), and it synchronizes the GSO replica server(s) with the GSO master server. This is not normally necessary because the GSO master server updates all available GSO replica server automatically on a regular basis. This task can be run only if this process needs to be forced to happen immediately, No parameters are necessary to run this task.

Recover Replica – This task is a more powerful variant of the synchronize replica task. It forces a GSO replica to completely rebuild its copy of the registry. The task should be run on a GSO replica server, for example, when the GSO master server was moved while this GSO replica server was offline or when the GSO replica server was inadvertently removed from the GSO cell by a Move Master Server task. The GSO cell name and the hostname of the GSO master server are required parameters for this task.

Reset Password – This task sets a new GSO cell password. It should be run on every GSO master and replica server. You would typically run this task if there is reason to assume that security has been compromised. A new GSO cell password is required as a parameter. Note that this task invalidates any cell backups taken before the password change. Note that this task should only be run in serial execution mode (the default is parallel).

Set Password Policy – This task allows you to define GSO cell-wide policies for GSO user passwords (not for GSO target passwords). This task must be run on the GSO master server. Several parameters can be provided: The

duration in days, hours, and minutes of the time a user can be logged on to GSO without refreshing the credentials and the lifespan (in days) and minimal length of a GSO password. By default, the lifespan of a user's GSO password is forever, specified by a value of 0 (zero) in the lifespan field.

The tasks described above are contained in the GSO Administration Tasks library (as shown in Figure 99 on page 205).

In addition to these administration tasks, the GSO Plus module comes with other task libraries that contain other tasks. Such tasks are, for example, provided for distribution of file packages or for routine preventive maintenance operations, such as cleaning up expired credential files on a server.

We recommend that you familiarize yourself with all the provided tasks because they offer excellent ways to simplify management and increase availability.

Another important management discipline for the management of distributed servers is monitoring system resources. GSO Plus offers a set of monitors that you can use to further automate the system's operation. Monitoring is covered in the section that follows.

7.2 Monitoring the GSO Servers

The IBM Global Sign-On for Multiplatforms, Version 2.0 comes with an extensive set of monitoring capabilities through its GSO Plus module. This allows for supervising GSO servers for their state and parameters, such as processes and disk space utilization, thus providing reliable information about the healthiness of the GSO environment. Once monitoring is activated, the GSO Indicator Collection, as part of the Tivoli desktop GUI, shows the state of the monitored resources through the use of thermometer gauges. The GSO monitoring prerequisites the Tivoli Distributed Monitoring Software on each system that is to be monitored. If the Tivoli Enterprise Console (TEC) is installed, TEC events can be used to automate the process of resolving the corresponding problem.

The status of the monitored resources can be viewed using the GSO indicator collections. The default GSO indicator collection is placed into the TivoliPlus collection upon installation of the GSO Plus module. In most cases, however, administrators prefer to place monitoring collections and indicator collections in separate Tivoli policy regions. This helps organize the desktop and delegation of responsibilities among administrators or if separation of authorization roles is a concern.

Figure 100 shows a practical example of a policy region that contains the GSO monitoring indicator collection and a profile manager for each of the following monitoring profile: GSO database monitor, GSO server monitor and GSO UNIX monitor.

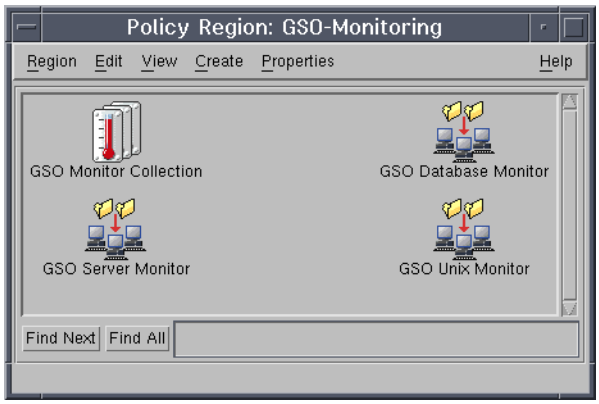


Figure 100. Customized GSO Monitor Profiles in a Policy Region

Remember, before you can add indicators to a policy region, that owning policy region must have the managed resources ProfileManager, IndicatorCollection, SentryProfile, and SentryProxy enabled.

Each of the monitoring profiles gets the corresponding subscribers, that is:

GSO Database Monitor: The GSO database servers
GSO Server Monitor: The GSO servers
GSO Unix Monitor: The GSO servers (UNIX only)

The monitors can be added to individual profile managers within policy regions with the copy function of the Tivoli Distributed Monitoring Profile Properties, as described, for example, in the *Tivoli/Sentry User's Guide*. However, the individual monitors may need to be adjusted to the actual implementation of the GSO servers. In particular, the filesystem names must correspond to the actual filesystem names which are created on the GSO server machines. If the default configuration is used for the GSO servers, the monitors supplied with GSO 2.0 can be used without modifications.

Monitors that are available with the GSO Plus module in the three monitoring collections as listed above cover monitoring of a great number of system resources. There are more than 50 separate monitors available that are preconfigured to monitor file and filesystem sizes as well as critical processes

on GSO server machines. Additional monitors can easily be added through the functions provided by Tivoli Distributed Monitoring.

Once the GSO indicator collection and the monitor profiles are created and customized within the desired policy region(s) and profile manager(s), the indicator collection can be set for the monitoring profiles, as shown in Figure 101, and the monitor profiles can be distributed to the respective system(s). By doing this, monitoring events will be shown in the specified indicator collection.

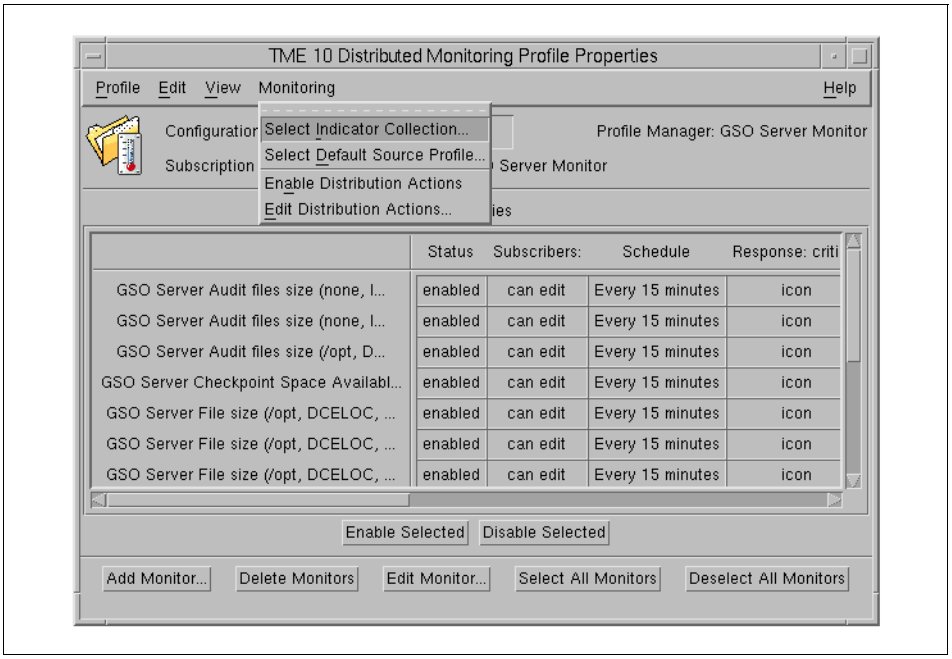


Figure 101. Setting the Indicator Collection for Monitor Profiles

After distributing the monitor profiles to the corresponding servers, the indicator collection (or profile status indicators) will indicate the healthiness of the GSO servers and services according to the definitions in the monitoring properties. The indicator on the thermometer rises as the status of a monitored resource becomes more urgent. By double-clicking on a monitor collection icon (also called a profile status indicator), a log of events related to that monitor profile is displayed, as can be seen in Figure 102.

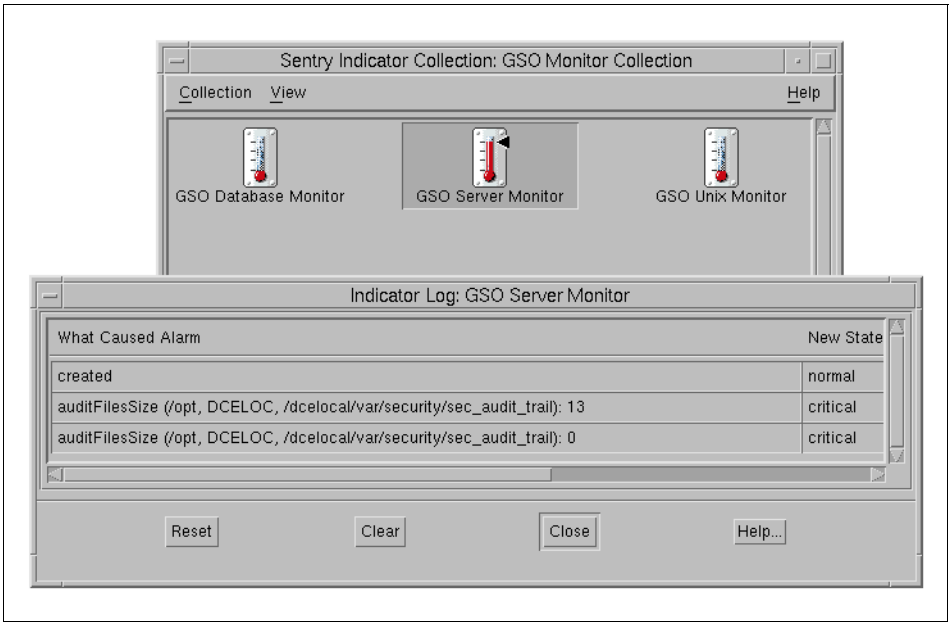


Figure 102. Displaying a Monitor Event

The Tivoli indicator collections give administrators an information about the state of resources. Monitor profiles can also be used to define automated responses. These possible responses can also include the execution of a user-specified task. However, for enterprise management and automation of event processing, the Tivoli Enterprise Console (TEC) should be used as explained in the next section.

7.3 Integration of GSO into Enterprise Event Management

The IBM Global Sign-On for Multiplatforms, Version 2.0 product provides an integration of GSO and DCE events into the Tivoli Enterprise Console (TEC). This enables the GSO management for a rules-based event management that integrates with network, systems, database, and application management. The integration offers a centralized, global view of the GSO environment. This enables the collection and automatic response to GSO events, such as a GSO server that is down, a filesystem that fills up or an event received from the DCE serviceability message that might indicate problems which GSO or DCE services. The Tivoli Enterprise Console allows for event prioritizing and correlation with its filtering and rule engine capabilities. Moreover, configurable administrator views allow for shared or

partitioned administrator responsibilities based on geography, organizational areas, resources, or other enterprise-defined areas of responsibility.

Figure 103 shows the flow how events from DCE serviceability logs and the distributed monitoring agents are processed by the event server and arrive at the event console.

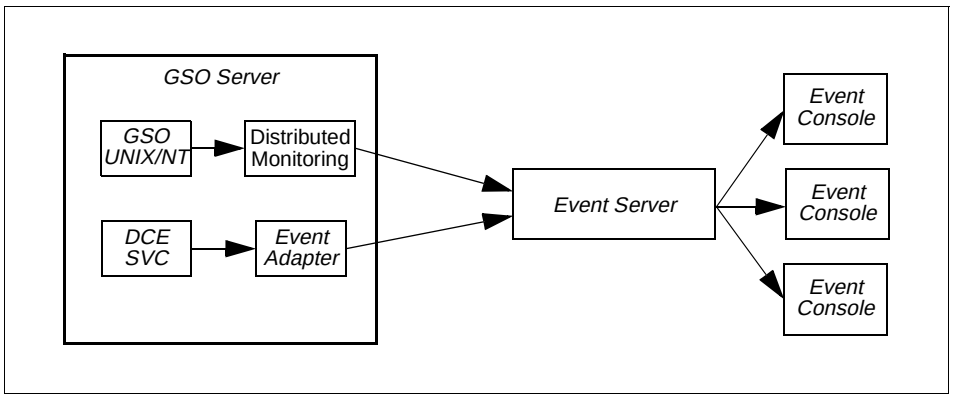


Figure 103. How GSO Events Get to the Tivoli Enterprise Console

7.3.1 Configuration of the Event Server for GSO

The integration of the GSO events into the Tivoli Enterprise Event Management is done by configuring the event server for GSO. Using the **Set Up Event Server for GSO** icon contained in the GSO Plus collection, the following activities are executed automatically:

- The GSO-related Event Group, *GSO 10Plus Events*, is loaded.
- Predefined rules, *Global Sign-On Rules*, are loaded which define responses to GSO related events. The configured rule set files are *gso.rls* and *dce_svc.rls*. These rules files contain specific actions, such as restarting GSO servers and cleaning up file systems. The rule files may be changed to add or modify event actions.

Figure 104 shows the configuration window for setting up the GSO event server that appears after you have double-clicked the **Set Up Event Server for GSO** icon and have selected a destination system. It is recommended to create a new rule base by cloning an existing rule base, thus preserving the current rule base.

The Path for New Rule Base field is used to specify a directory on the event server. The administrator needs write access to the specified path.

It is optional to specify a Name of Event Console to Configure to display GSO related events on a particular system administrator's event console. The name that appears under the desired system administrator's event console icon has to be entered into the configuration menu. The default value for the installation path is /usr/tec_rules/GLOBAL_SIGNON. Prior to the installation process, the available disk space should be checked.

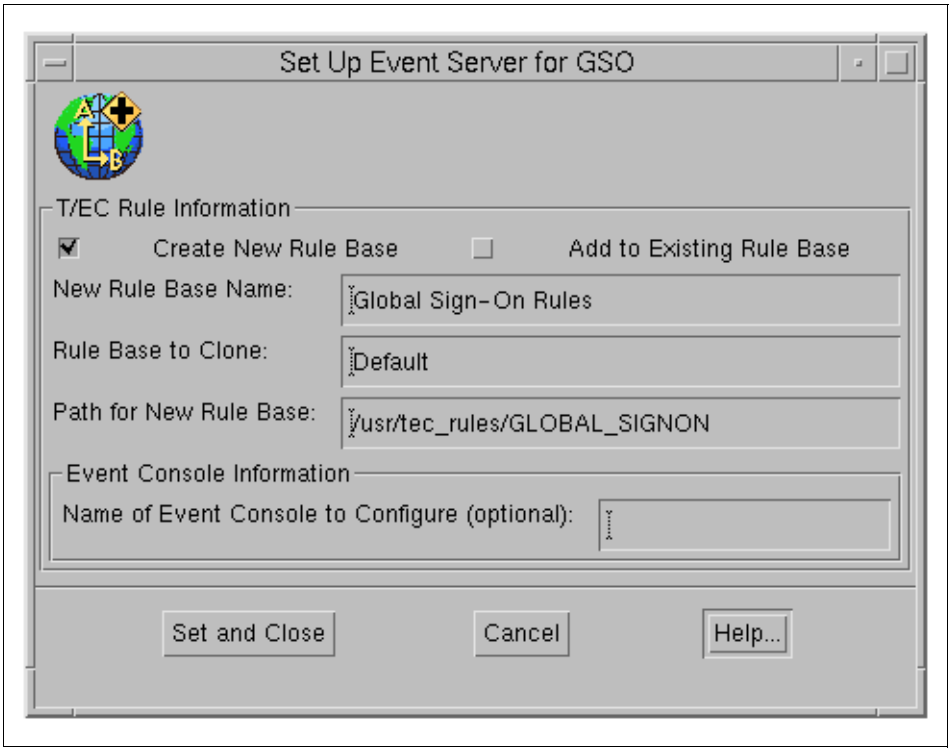


Figure 104. Set Up the Event Server for GSO

The successful configuration of the event server can be tested by creating a GSO event and verifying its reception by the event server. In order to create a GSO event, for example an event from the *DCEServiceability* class, the command line interface `wpostemsg` may be used:

```
wpostemsg -r HARMLESS -m 'This is a test message' SVCEvent DCEServiceability
```

The event should be displayed on the Tivoli Enterprise Console (TEC) as shown in Figure 105.

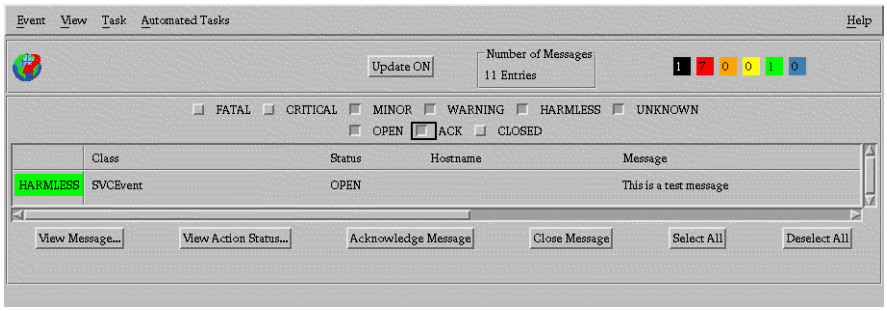


Figure 105. GSO Plus Event in Tivoli Enterprise Console

Alternatively, one can check whether the event has been successfully sent to the TEC by issuing the command:

```
wtdumprl
```

The output should look something like:

```
PROCESSED

1~2125~0~904808834(Sep 03 02:47:14 1998)
### EVENT ###
SVCEvent;
source=DCEServiceability;
severity=HARMLESS;
msg='This is a test message';
origin=9.3.1.68;
END

### END EVENT ###
```

Use the command `wtdumper` to see if the event has been fully processed by the TEC daemons. The output should look something like:

```
wtdumper -d
SVCEvent;
        server_handle=1;
        date_reception=904808954;
        event_handle=1;
        source=DCEServiceability;
        sub_source='';
        origin=9.3.1.68;
        sub_origin='';
        hostname='';
        adapter_host='';
        status=OPEN;
        administrator='';;
        acl=[ admin];
        severity=HARMLESS;
        date='Sep 03, 1998 02:49';
        duration=0;
        msg='This is a test message';
        msg_catalog='';
        msg_index=0;
        num_actions=0;
        credibility=0;
        repeat_count=0;
        cause_date_reception=0;
        cause_event_handle=0;
        svc_component=unknown;
        svc_msg_ID=unknown;
        svc_process_ID=unknown;
        svc_severity=unknown;
        svc_src_file=unknown;
        svc_src_line=0;
        svc_sub_component=unknown;
        svc_thread_ID=unknown;
        svc_time=unknown;
END
```

The Tivoli event server will now receive and process events that are sent from the distributed monitoring. Table 22 lists events that could arrive on the Tivoli Enterprise Console (TEC) as GSO Plus events.

Table 22. GSO-Related TEC Events

Event Class	Source
GSOServerUp GSODiskSpaceUsed GSOServerFileSize GSOAuditFilesSize GSOCheckpointSpaceAvailable universal_swapavail Sentry2_0_inodesusedpct	Distributed Monitoring

Event Class	Source
SVCEvent DCEPotentialSecViolationAttempt DCEInsufficientMemory GSORootSignonFailureAttempt	DCE Serviceability

The GSO documentation contains additional information on the specific-event-correlation processing of these events.

7.3.2 Configuration of GSO Event Adapters

GSO uses DCE as the underlying security and persistent store services. The DCE infrastructure comes with a DCE serviceability mechanism that is designed to be used mainly for server informational and error messaging—. that is, for messages that are of interest to those who are concerned with server maintenance and administration. The essential idea of the mechanism is that all server events that are significant for maintaining or restoring normal operation should be reported in messages that are made to be self-explanatory. DCE servers, such as the DCE security server and the DCE cell directory server, as well as the GSO servers, use DCE serviceability messages to provide fatal, error and warning messages to an administrator.

As shown in Figure 103, the GSO event adapter sends these DCE serviceability messages to Tivoli event server, thus making these available to Tivoli administrators through the Tivoli Enterprise Console or to automate event processing. The GSO event adapter has to be configured on every system where DCE serviceability logs have to be monitored. Therefore, one typically will configure the GSO event adapter on the GSO master server and on every GSO replica server.

The configuration of the GSO event adapter takes place by executing the Configure GSO Event Adapter task as shown in Figure 106 (double-click on the icon).

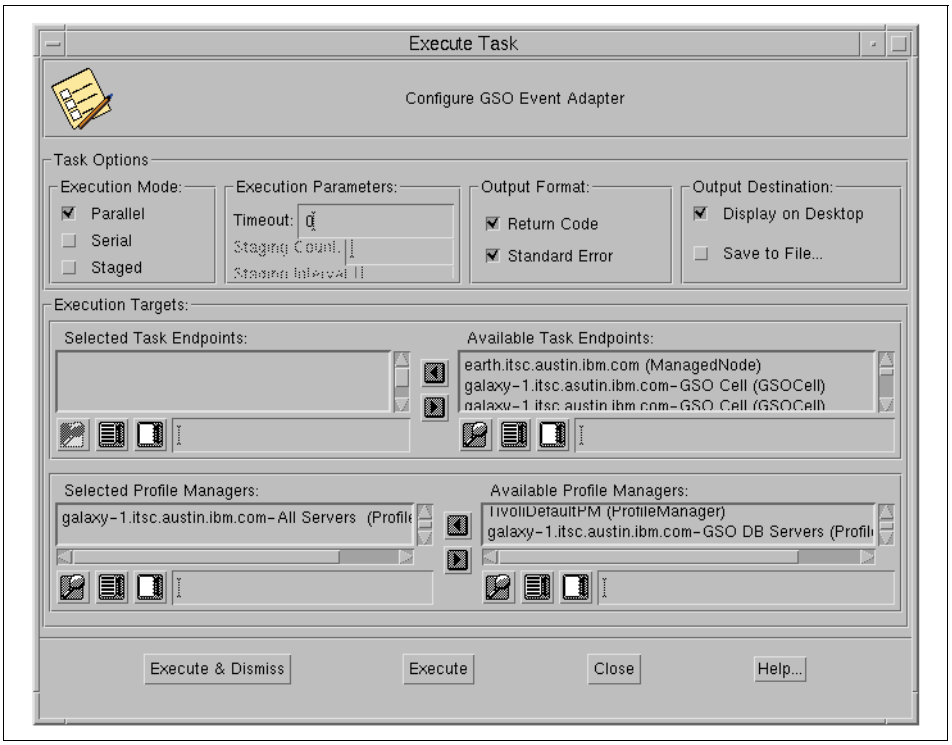


Figure 106. Configuration of Event Adapters

The configuration task modifies the DCE serviceability routing file, /opt/dcelocal/var/svc/routing, and starts the GSO event adapter daemon on the specified GSO server. The entries in the generated routing file are:

```
#
#Global Sign-On2.0
FATAL:BINFILE:/usr/lpp/dce/var/svc/bin.log;STDERR:-;FILE:/opt/dcelocal/var/svc/fatal.
log
ERROR:BINFILE:/usr/lpp/dce/var/svc/bin.log;STDERR:-;FILE:/opt/dcelocal/var/svc/error.
log
WARNING:BINFILE:/usr/lpp/dce/var/svc/bin.log;STDOUT:-;FILE:/opt/dcelocal/var/svc/warn
ing.log
#NOTICE:BINFILE:/usr/lpp/dce/var/svc/bin.log;STDOUT:-;FILE:/opt/dcelocal/var/svc/noti
ce.log
#NOTICE_VERBOSE:BINFILE:/usr/lpp/dce/var/svc/bin.log;STDOUT:-;FILE:/opt/dcelocal/var/
svc/verbose.log
#
```

As indicated, messages with the severities Fatal, Error and Warning are written to the binary file, /usr/lpp/dce/var/svc/bin.log, which is used by the GSO event adapter process, `dce_tecad`. Moreover, DCE serviceability

messages are written to STDERR and to human-readable log files, for example /opt/dcelocal/var/svc/fatal.log.

The corresponding files grow over time and may fill up the file systems. Therefore, a periodical cleanup should take place which removes all log files.

Once the GSO event adapter is configured, DCE serviceability events will generate Tivoli events that are processed by the Tivoli event server. The Tivoli administrator can view the GSO-related events through the Tivoli Enterprise Console (TEC) as shown in Figure 105 on page 215.

7.4 Auditing

In order to prove that protected resources are only used according to their security standards, most organizations demand logging of successful and unsuccessful attempts to access systems or to access protected resources on a system by means of audit records. Moreover, access violations—that is unauthorized access attempts to systems or information—have to be recognized as violations, either immediately or on subsequent analysis.

To provide the necessary audit records for a GSO installation, audit records from the following systems have to be collected:

- Tivoli Framework
- Server Operating System
- DCE Components, for example the DCE security server
- GSO Components, for example the GSO server

The Tivoli Framework as well as the server operating systems provide the ability to generate audit records. For example, a description on how to configure the AIX audit subsystem can be found in the *System Management Guide: Operating System and Devices* for the corresponding AIX release. Moreover, the Tivoli Security Management product, which provides a solution for role-based distributed security management, also provides support for auditing, including audit reports. If the Tivoli Enterprise Console (TEC) is installed, the audit messages can be forwarded for processing and correlation.

The Distributed Computing Environment (DCE) comes with a audit subsystem that can be used to collect security-relevant events from the DCE security server, the time services and the audit subsystem. The GSO daemon (gsod) also uses the DCE audit services to record its events. The following section describes the configuration of the DCE audit service.

DCE Integration with TEC

At the time of writing this book, a beta version of the IBM DCE Tivoli Management Framework code was available that integrates not only DCE management operation into the TME but also allows forwarding and handling of DCE audit events by the Tivoli Enterprise Console.

For the latest status on this, including more information, check the following Web link: `www.software.ibm.com/enetwork/dce`.

The DCE audit services comes with an audit daemon (auditd), which performs the logging of audit records bases on specified criteria. DCE applications, for example the DCE security server (secd), use the audit application programming interface to record audit events. The administrative command interface, dcecp, is used to configure the audit service.

7.4.1 Enabling and Configuring the DCE Audit Service

To enable the DCE audit service, the DCE audit daemon has to be configured. This can be done through the command interface config.dce (formerly mkdce):

```
config.dce audit
```

The auditd daemon should be configured at least on all hosts where GSO servers are configured.

To enable servers, for example the DCE security server or the GSO daemon, the environment variable

```
DCEAUDITON=1
```

has to be set prior to starting these servers. In order to restrict the sizes of the audit trail files, it may be appropriate to set the following environment variables:

```
DCEAUDITTRAILSIZE=4194304 # Max. audit trail file size is 4 MB
DCEAUDITWRAP=1 # wrap around as soon as the space limit is reached
```

It is good practice to include these environment variables into the /etc/environment file (UNIX).

By using audit filters, it is possible to control the amount of audit information; see the corresponding DCE documentation for details.

7.4.2 Displaying DCE and GSO Audit Information

Audit records can be displayed by using the dcecp command on the audit trail files:

```
dcecp -c audtrail show /opt/dcelocal/var/security/sec_audit_trail
dcecp -c audtrail show /opt/dcelocal/var/audit/adm/central_trail
dcecp -c audtrail show /opt/dcelocal/var/adm/time/dts_aud_trail
dcecp -c audtrail show /var/gso/gsod/audit.log
```

The sec_audit_trail file contains audit events from the DCE security server related to:

- Authentication and cryptographic events
- Access control, security state modification and query, lookup or test events (for example adding/deleting accounts), replacement of access control lists, lookup for ERA (Extended Registry Attributes)
- Server configuration and administration events, for example initialization of security replicas

The central_trail file contains events related to the DCE audit service itself; the dts_aud_trail file contains information about the DCE time service (DTS).

The GSO-related audit trail file, /var/gso/gsod/audit.log, contains events related to the GSO daemon process (gsod).

The GSO daemon records the events related to GSO account management, including the modification of target information.

Example

The following example shows GSO audit information that is generated when a GSO user is added by using the Tivoli User Administration.

The Tivoli notification service provides the following information:

```
Date: Fri Oct 23 06:25:25 CDT 1998
Notice-Group-Name: User Management
Priority: Notice
Sent-By-Administrator: junior@mars.itsc.austin.ibm.com

The user William Schulz has been created in the profile GSO-Admins.

Notice-id: 0
Date: Fri Oct 23 06:32:05 CDT 1998
Notice-Group-Name: User Management
Priority: Notice
Sent-By-Administrator: junior@mars.itsc.austin.ibm.com

The distribution of profile GSO-Admins to
galaxy.itsc.austin.ibm.com-GSO Cell has succeeded.
```

The GSO audit trail contains the following record that shows that the request was done by gso_tme_admin, which is the DCE principal used by GSO:

```
--- Event Record number 846 ---
o Event Information:
   - Event Number:        0x81000000 /* -2130706432 */
   - Event Name:          Unknown
   - Event Outcome:       success
o Server:                 /.:/hosts/earth.itsc.austin.ibm.com
o Client:                 /.../galaxy.itsc.austin.ibm.com/gso_tme_admin
o Number of groups:       4
   - Group 0:             /.../galaxy.itsc.austin.ibm.com/none
   - Group 1:             /.../galaxy.itsc.austin.ibm.com/none
   - Group 2:             /.../galaxy.itsc.austin.ibm.com/gso-sr-admin
   - Group 3:             /.../galaxy.itsc.austin.ibm.com/gso-user
o Authorization Status:   Authorized with a pac
o Date and Time recorded: 1998-10-23-06:31:43.305-05:00I-----
o Client Address:         ncadg_ip_udp:10.1.1.1[33569]
--- End of Event record number 846 ---
```

Entries about adding the DCE related information to the DCE security server, for example adding the corresponding principal, account and extended registry attributes, can be found in the sec_aud_trail file:

```
--- Event Record number 646 ---
o Event Information:
   - Event Number:           0x114 /* 276 */
   - Event Name:             PGO_Add
   - Event Outcome:          success
o Server:                    /.:/hosts/earth.itsc.austin.ibm.com
o Client:                    /.../galaxy.itsc.austin.ibm.com/gso_server
o Number of groups:          0
o Authorization Status:      Authorized with a pac
o Date and Time recorded:    1998-10-23-06:31:43.501-05:00I-----
o 2 Event(s) specific:
   - item number 1           long int 0
   - item number 2           char string wschulz
--- End of Event record number 646 ---
(Events 647 to 654 not shown)
--- Event Record number 655 ---
o Event Information:
   - Event Number:           0x12b /* 299 */
   - Event Name:             ERA_Update
   - Event Outcome:          success
o Server:                    /.:/hosts/earth.itsc.austin.ibm.com
o Client:                    /.../galaxy.itsc.austin.ibm.com/gso_server
o Number of groups:          0
o Authorization Status:      Authorized with a pac
o Date and Time recorded:    1998-10-23-06:31:48.008-05:00I-----
o 3 Event(s) specific:
   - item number 1           char string principal/wschulz
   - item number 2           ulong int 1
   - item number 3           uuid info
7d600926-5765-11d0-8285-0004ac605597
--- End of Event record number 655 ---
```

As the `Client` field indicates, the requester for the DCE security registry updates is the gso_server principal, which is the DCE principal used by the GSO daemon process.

Chapter 8. Managing User Accounts

Managing user accounts is likely to be the most common administration job necessary in a production GSO environment. It includes adding and deleting user accounts and changing existing account records, such as resetting a user's forgotten GSO password.

This chapter describes how GSO user administration works and how typical tasks are performed. User administration for GSO also includes defining targets for users. This chapter, however, does not elaborate on how target information is being added to user records because this was described in 6.3, "Adding Targets for Users" on page 161.

8.1 Integration in Tivoli User Administration

User accounts for GSO 2.0 are integrated into the Tivoli User Administration application. Thus, all administration tasks are done through the Tivoli User Administration application, either by using the GUI or by using the command line interface (CLI).

The following is a brief recap of the Tivoli User Administration (UA) to better understand how GSO user accounts are managed in the Tivoli Management Environment (TME).

The TME is, to a large extent, platform-independent, and thus the Tivoli UA is designed to support management of user accounts for a variety of different platforms, such as Windows NT, UNIX, Novell NetWare, RACF (mainframes), and others. These UA-supported platforms should not be confused with GSO targets, although they might actually be the same systems, as for example Novell NetWare. While Tivoli UA provides functions for adding and deleting user accounts in NetWare (and all the other supported platforms) GSO only maintains a record of user accounts for these platforms with their related passwords for the purpose of user authentication. Figure 107 depicts the relationship between Tivoli UA, GSO and other Tivoli UA platforms, and it is discussed in the following section.

8.1.1 GSO as a Managed Resource

On the left-hand side, Figure 107 shows a database that is part of Tivoli UA. It contains all information about user accounts that Tivoli UA manages. Most administration operations, such as adding or changing user accounts, directly affect the data in this database. On the right-hand side, every supported platform stores some kind of user information, too. For example, a UNIX

machine stores its user records in files, such as /etc/passwd and /etc/group. Tivoli UA manages user accounts through so-called *user profiles*. User profiles are managed from within *profile managers*. The next section in this chapter describes user profiles and profile managers in more detail.

After changes are being applied to the Tivoli UA database, the profile(s) need to be *distributed* to the destination platform in order to become effective. This is an asynchronous process that might be initiated any time when necessary, for example after a number of new user accounts have been added to the Tivoli UA database.

The reverse process to distribution of a profile is called *populating* a profile. When populating a profile, Tivoli UA "learns" about the user account information that already existed on the destination platform(s). Distribution and population is the subject of section 8.1.3, "Distribution and Population of User Profiles" on page 230.

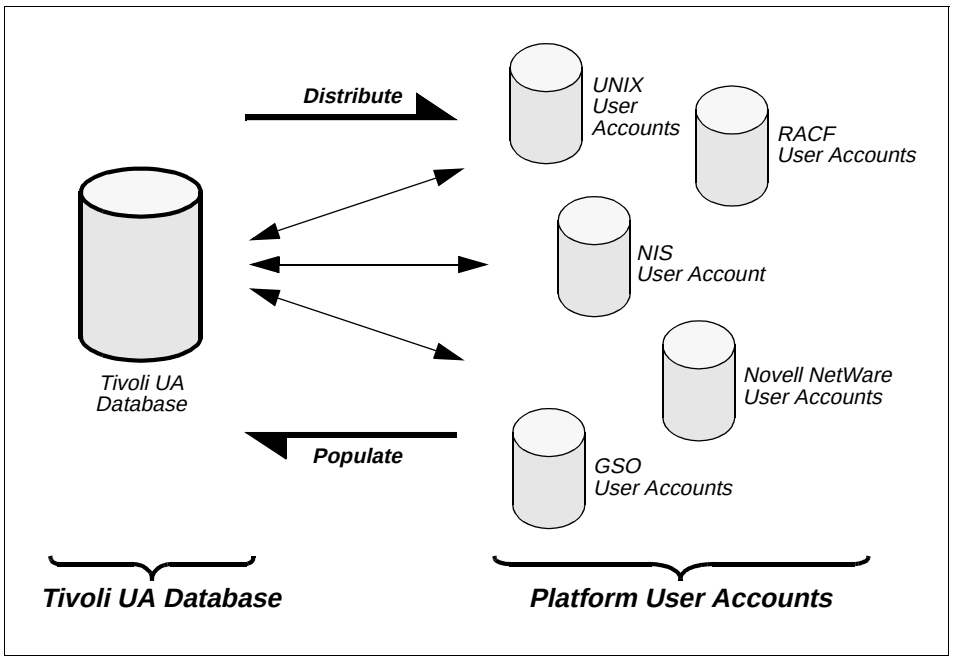


Figure 107. Tivoli UA Database Versus Platform User Account Information

GSO user accounts are managed in Tivoli UA exactly the same way as all the other user accounts. In fact, Tivoli UA does not distinguish between GSO user accounts and any other user accounts, but there certainly are attributes that are valid for particular platforms, such as GSO, only.

The Tivoli User Administration application, when installed from the product media, does not know about GSO. Only when the TME 10 GSO User Administration module is installed (see 4.3.1, “Installing the TME 10 GSO User Administration” on page 71), is Tivoli UA being extended by the functionality that is necessary to support GSO user accounts. Figure 108 was introduced in 2.7, “Integration with Tivoli” on page 30, and it shows you how TME 10 GSO User Administration extends the Tivoli UA capabilities in order to support GSO user accounts. The most visible addition to Tivoli UA is the additional support for the GSO-specific attributes in the Tivoli UA GUI.

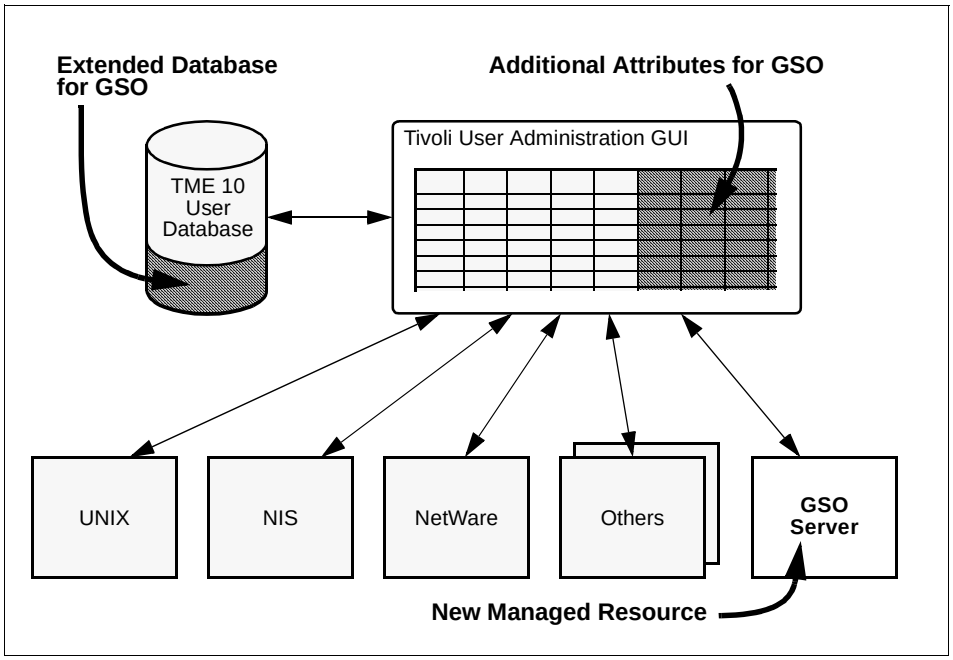


Figure 108. GSO User Administration Extensions to Tivoli User Administration

Upon distribution of a user profile, the GSO master server, shown in the lower-right corner of Figure 108, receives that information from Tivoli UA. Thus, from the Tivoli UA point of view, GSO is just a managed resource that receives user profile information when a user profile is distributed to it, much the same as others, such as UNIX or Novell NetWare. The GSO master server, in turn, stores and provides this information to GSO replica servers and the clients such that users can log on to GSO, and GSO can authenticate them to the targets without the direct involvement of the Tivoli UA.

8.1.2 User Profiles in Tivoli User Administration

User account information in Tivoli UA is organized in user profiles. There can be several user profiles or just a single one that stores all information, for example, an organization's employees. Thus, a user profile contains records of multiple users that are treated as a unit when distributing them to a managed resource, such as GSO.

Though this book is not meant to explain features that are common to all Tivoli applications, such as organizing policy regions and profile managers, it might be worth it to recap some of the concepts. A *policy* is a set of rules that are applied to managed resources. A *policy region* is a collection, or container, for resources that use the same set of policies. For example, a policy region might be assigned to an administrator who is then authorized to perform certain tasks to the resources contained in that region. A policy region could also represent a physical or logical part of a whole infrastructure for which common rules apply. A *profile* is a set of information pertinent to a specific application or other resources, such as users. A *profile manager*, as such a managed resource usually contained in a policy region, is a place to create and organize groups of profiles and link recipients to them. The information stored in profiles is only a copy of the actual information that resides on whatever that profile represents. Profiles need to be distributed to the recipients (also called *subscribers*) in order to make the information they contain active. Population is the reverse process where a profile is filled in by information gathered from an actual system. Profile managers can be organized in hierarchies such that one profile manager becomes the subscriber of another. This is done, for example, to give a senior administrator a higher level of authority by letting him or her control subsequent profile managers while these lower-level profile managers only control a part of the whole.

The way policy regions, profile managers and profiles are organized depends heavily on the individual environment. In a small organization or in a test environment, as little as one of each could be sufficient, while a large organization will certainly distribute administrator responsibilities, workload, and physical or logical boundaries to several policy regions with multiple profile managers. Defining these boundaries and organizing the Tivoli environment is an important task when the Tivoli Management Environment is being deployed in an organization.

For more information about the TME concepts, we refer you to the various Tivoli redbooks in the *Tivoli Redbooks Collection* CD-ROM (see F.2, "Redbooks on CD-ROMs" on page 293, for ordering information) and to the

Tivoli product documentation. Please also review section 3.7, “Tivoli Integration” on page 48, in this book.

Administration of single users is done through the User Properties dialog window as shown in Figure 109. It opens when you double-click on a user record in a user profile. The left-most portion of that dialog lets you choose from a list of categories, and the properties belonging to that category will then be displayed to the right of that list. GSO is such a category, and it has already been chosen in the example shown in Figure 109.

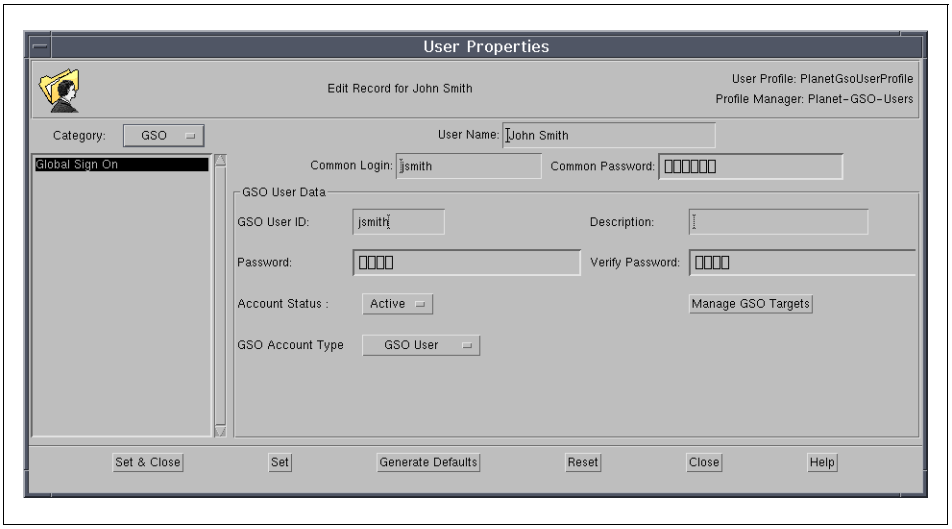


Figure 109. GSO User Properties

It should be noted that the same user might exist in other categories as well, meaning that this user has a user account not only in GSO but also on other platforms. In fact, it is very likely and even recommended that the same user is administered with Tivoli UA for all the platforms that person will be working with. This way, a user is created and administered with one single management tool for multiple platforms, and GSO later takes care of the user authentication on these platforms.

In addition to the information supported and stored by the Tivoli UA, GSO expands this by the following (see Figure 109 and subsequent sections in this chapter):

GSO User ID	The user ID with which the user logs on to GSO.
Description	An information field for any user information.
GSO Password	The password required from the user to log on to GSO.

Account Status	Either *active* or *inactive*, specifies whether or not this user is enabled to log on to GSO.
Account Type	Either *GSO User* or *GSO Sec Officer*. Specifies whether this user is a normal user or a security officer (a GSO senior administrator) authorized to do user administration tasks such as enabling Smartcard support (see 5.4.2, “Setting Up and Using GSO Smartcard Administration” on page 132).
Target Information	Any target information for that particular user. Please see section 6.3, “Adding Targets for Users” on page 161, for more information.

8.1.3 Distribution and Population of User Profiles

As mentioned above, a user profile in Tivoli is only a copy of the actual user information for administrative purposes (see Figure 107 on page 226). It is stored in the TMR (Tivoli Management Region) server’s database. Depending on the platform, the actual user information may be stored in a Novell NetWare server, in GSO servers, or in any other supported platform(s). Administrative operations, such as add, delete, and change, only affect the copy of the user profile in the TMR server and therefore do not take effect immediately on the target platform.

In order to become effective, a user profile needs to be distributed to the supported endpoint (or platform). GSO is such an endpoint, and any changes to user data using the Tivoli UA GUI only take effect after the user profile has been distributed to the GSO cell. Distribution can happen in various ways, either manually upon initiation or automatically at scheduled times and intervals. See the appropriate Tivoli documentation for more details about this.

Figure 110 shows an example Distribute Profile window that you might get when a manual distribution is being initiated. As the profile name in this example suggests, this user profile stores user account information for GSO and Windows NT user accounts. Take note that the list on the left, the Distribute to These Subscribers list not only contains the GSO cell as a recipient of that user profile (the first entry in the example) but also lists some Windows NT systems because they need to be updated with this user profile information, too.

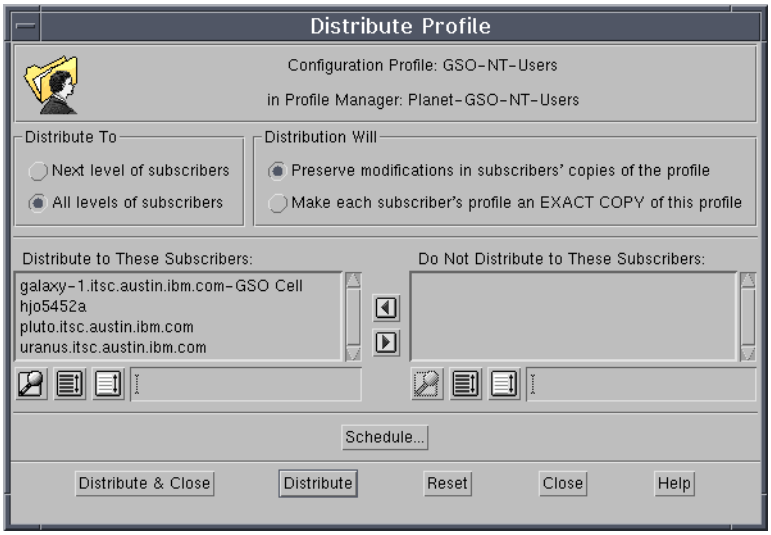


Figure 110. Profile Distribution Options

When planning for distribution of user profiles, the following should be considered:

- The endpoint of the distribution of GSO users must be the GSO cell. Make sure that “All levels of subscribers” is selected in the Distribute To block (Figure 110), unless you speicifically don’t want this to happen.
- Because during a distribution users may encounter temporary delays or interruption of the login service, distribution should be scheduled during off-hours.
- User passwords are not reset during a distribution because they might have been changed by the user. Only if a new password was entered for a user, will this password replace the current one in the GSO cell.
- The option to make a subscriber an exact copy of the user profile is not supported in GSO. (This option, if supported and selected, would cause any user account information in the GSO cell that is not defined in a profile to be deleted upon distribution. It would also require that all user information is managed from one single user profile since every distribution process would delete all other user accounts.)

The reverse process to distribution is called *population*. During population, user account information is gathered on the selected target platform and stored in the user profile. This is most often used to initially populate an empty user profile.

Population of GSO user profiles is not supported through the GUI, but the command line interface (CLI) can be used instead. You must run the following command to populate a user profile:

```
wpopusrs [-o] -l @GSOCell:<gso cell name> @UserProfile:<user profile name>
```

The `-o` flag causes any existing user information stored in the user profile to be overwritten. The `-l` flag specifies that users' home directories remain untouched. It is a required flag for the `wpopusrs` command, although it does not really apply for GSO user accounts (there are other options for the `wpopusrs` command that affect the users' home directories, but they are not applicable for GSO user accounts).

Population without the `-o` flag (see command above) does not affect existing users in the profile, and an advisory message will be printed for every user that could not be added to the profile because it already existed.

It should be noted that populating GSO user accounts from a GSO cell cannot gather all information about the users. The common name and common user ID, for example, are set to the GSO user ID, because the `wpopusrs` command cannot determine their actual values.

Note on Populating Profiles

You should always be very careful when using both the populate and the distribute functions as part of regular processes. It is generally a good idea to declare the Tivoli user profiles as the master source of user information, and thus, any populate operation would compromise this idea.

For example, if, for any reason, the population of a profile adds users that you consider are not GSO users, you might be tempted to delete these users from the profile. Upon next distribution, these users will be deleted from the GSO cell as well, which might not have been your intention.

8.1.4 Administrator Roles for User Administration

Tivoli supports separate administrator roles that have different authorizations to perform administrative tasks. The base roles in Tivoli are *user*, *admin*, *senior*, and *super*. An administrator has one of these roles assigned to him or her.

The two Tivoli administrator roles that are important for GSO user administration are admin and senior. An administrator with the admin role can add, change, and delete user information as well as target information for

individual users. A senior administrator can, in addition, manage passtickets, target groups and he/she can create other administrators with the admin role.

Section 3.9.3, “Tivoli GSO Admin Security” on page 56, contains additional information about Tivoli/GSO administrator roles.

8.2 Adding, Changing and Deleting User Account

Daily administration of GSO user accounts includes add, change, and delete operations that are performed using the Tivoli User Administration GUI. Using this GUI, these operations are fairly easy and intuitive, as explained in the sections that follow.

When adding a GSO user account, there are generally two options:

- A new user account needs to be added; that is, this account did not exist before at all.
- An existing, non-GSO user account needs to be added to the GSO cell;, that is, that user might have existed already on other platforms, but is not yet added to the GSO cell.

In both cases, for the particular user account that is being added, the correct policy region, profile manager and user profile needs to be identified by the administrator. The organization of these resources depends on how Tivoli is structured in order to reflect your organization’s needs (see the discussion in the first sections of this chapter).

Once the user profile is identified and opened, the User Profile Properties window, like the example shown in Figure 111, is presented to the administrator.

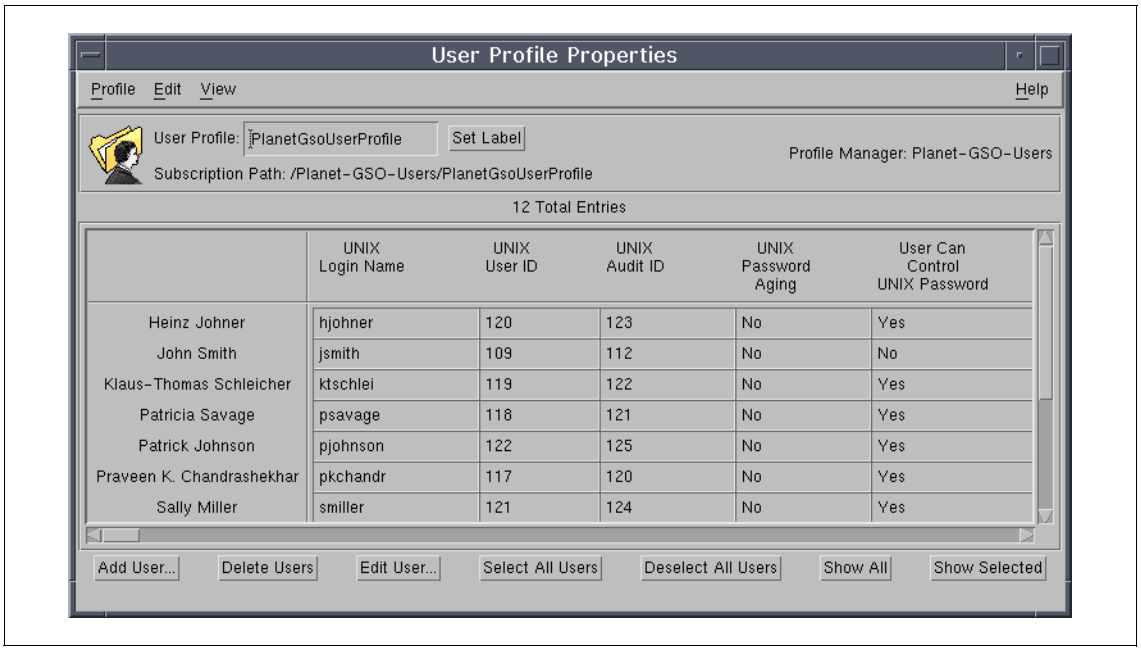


Figure 111. User Profile Properties

Quite obviously now, clicking on either **Add User...** or **Edit User...** (after selecting the appropriate user from the list) opens the User Properties dialog window that was introduced in Figure 109 on page 229. Depending on the action chosen (add or edit), the fields will be blank, or they contain the information for that particular user. The category list to the left allows you to select the set of properties that will be shown on the rest of the window.

A user might have properties in several categories. For example, when a user account is defined for Windows NT, Novell NetWare, and GSO, there will be information required in all these categories. As far as GSO is concerned, at least the *Identification* and *Global Sign On* categories are of interest for an administrator. The former specifies some general properties for each user account, such as user name, location, telephone number, office number, and so on.

When the Global Sign On category is selected in the list to the left (Figure 109 on page 229), the right-hand view of the User Properties dialog will look like the one shown in Figure 112 (the left portion of this window has been cropped to have a larger view).

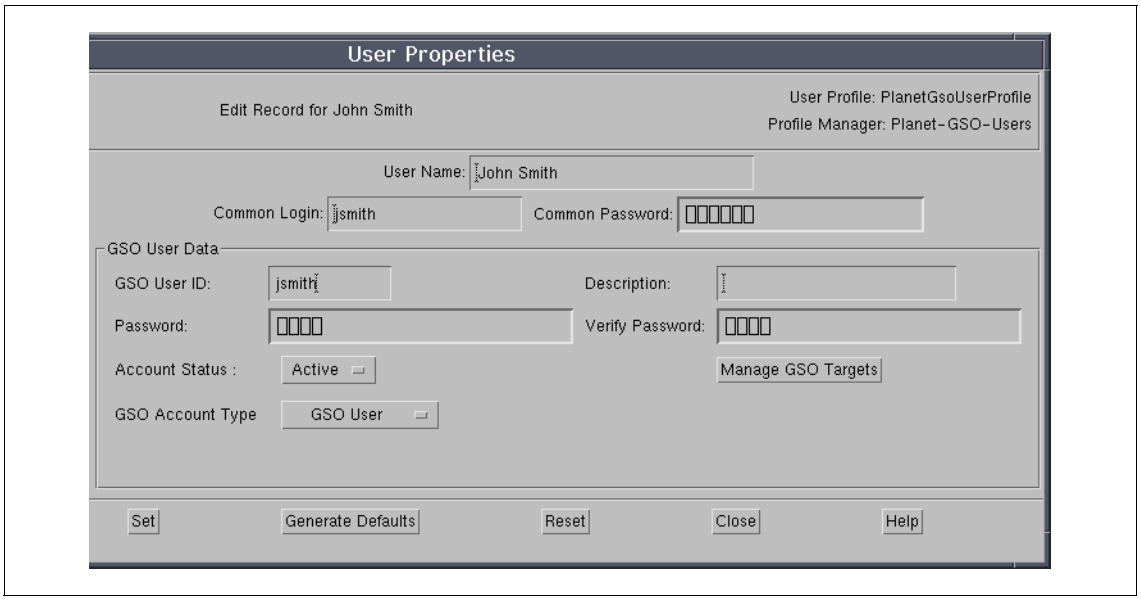


Figure 112. User Properties Specific to GSO

The fields in the upper parts, the User Name, Common Login and Common Password fields, are common to all categories, and they are usually entered in the Identification category view (though they could be entered and/or changed in the GSO category view as well).

The information shown in the GSO User Data block (Figure 112) is what can (or needs to) be defined for GSO user accounts; please read the description to Figure 109 on page 229 for more details. GSO target definitions for users are described in 6.3, “Adding Targets for Users” on page 161.

In order to delete a GSO user account, select (highlight) its record in the User Profile Properties window (Figure 111) and click on **Delete Users**. Multiple users can be selected for one delete operation.

Do Not Forget to Distribute!

As explained earlier, any changes to the user profiles affect the TMR server’s database only, and GSO as an endpoint will only be updated when the user profile is being distributed.

Thus, do not forget to distribute the user profile to the GSO cell after any changes to make them effective.

8.3 GSO Password Reset

When a new GSO user is added, the administrator will most likely add a GSO password to the user's profile record. This will be the user's initial GSO password. Because the user may choose to change his/her password using the GSO user interface on the workstation, subsequent distribution of the user profile will not reset the user's GSO password.

If a user has forgotten his or her GSO password, the administrator can specify a new password on the GSO user properties panel shown in Figure 112. Only when it has been changed on this panel will it be reset in the GSO cell when the profile is distributed. This way, the user as well as an administrator can change that user's GSO password.

Note on GSO Passwords

As mentioned earlier in this book, GSO uses the Distributed Computing Environment (DCE) for authentication and other purposes. A user's GSO password is actually his/her DCE password.

Because of this, rather than editing user profiles and distribute them to the GSO cell to reset a user's forgotten password, it could also be reset using standard DCE management methods, such as the `dcecp account modify` command or on AIX by using SMIT with the `chpass` fastpath.

Appendix A. Extended Configuration Methods

IBM Global Sign-On for Multiplatforms, Version 2.0, is designed to be installed and configured in an environment that has to meet certain prerequisites as described throughout this book and in the IBM product announcement letter. Some customer environments, however, may not meet all these prerequisites. For example, there might be a DCE cell already in place that the customer wishes to utilize rather than to create a new DCE cell for GSO. This appendix provides additional information on how GSO can be configured using an already existing DCE cell and how to recover from a failed configuration. Some other reference information is also given throughout this appendix.

In order to understand the configuration into an existing DCE cell, it is assumed that the reader has a good understanding of DCE.

A.1 Configuration of GSO Servers on an Existing DCE Cell

When installing GSO servers using the provided methods as described in Chapter 4, “Installing GSO Servers” on page 61, Tivoli management tasks are being used that do all the work, including the creation of a new DCE cell. If there is already a DCE cell configured, these management tasks terminate without any action, thus preserving the existing DCE configuration.

The installation of GSO into an existing DCE cell can either be done by the provided command line interface or by creating new Tivoli management tasks that do not create a new DCE cell. This section explains how to configure GSO through the command line interface.

A.1.1 DCE Prerequisites

To install GSO into an existing DCE cell, the following prerequisites have to be met:

- The DCE client services are installed and configured on the machine where a GSO server is to be installed. Note that although the GSO servers seem to directly communicate with the DCE servers, this is in fact accomplished through DCE clients, even though this all might be on a single machine.
- The DCE Threads Compatibility Library is installed.
- The DCE Privacy Level Protection Feature is installed.

- A DCE account with sufficient privileges to create and modify objects in the security and directory name space is available. Usually this is the DCE account `cell_admin`.

For your reference, Table 23 lists objects which are created or modified in the DCE cell during the configuration process.

Table 23. DCE Objects Created or Codified by GSO

Object Type	Name
Groups	gso-sr-admin
	gso-admin
	gso-mts
	gso-user
	intraverse/dsb-servers
	gsodb_group
Accounts & Principals	gso_server
	gso_tme_admin
	intraverse/dsb/default/<hostname>
	gsodb
	cell_admin Remark: The access control list for `cell_admin` account is modified to grant `rcDnfmaug` to gso_server. If an other account is specified during the configuration process, it will be used accordingly.
Security Schema	/.:/sec/xattrschema/gso_um
	/.:/sec/xattrschema/gso_gim
	/.:/sec/xattrschema/gso_pkm
	/.:/sec/xattrschema/gso_ptkt
	/.:/sec/xattrschema/gso_policy_rgy_mgmt
	/.:/sec/xattrschema/gso_policy_tivoli_mgmt
	/.:/sec/xattrschema/gso_policy_group
	/.:/sec/xattrschema/gso_policy_org
CDS Entries	/.:/gso-cfg-servers
	/.:/gso-keyed-servers
	/.:/gso-servers
	/.:/hosts/<hostname>/gso-server
	/.:/hosts/<hostname>/gso_master
	/.:/hosts/<hostname>/gso_replica
	/.:/subsys/gso
	/.:/subsys/gso/db

A.1.2 Installation of the GSO Software

Before the configuration of the GSO components can take place, the GSO software has to be installed on the server and client systems. The software installation process can be done by the Tivoli software installation and the provided GSO file packages as described in 4.4, “Installing Servers Using Tivoli Software Distribution” on page 76, or by other means, such as the `installp` command on AIX. Table 24 and Table 25 list the GSO software which has to be installed on servers and clients, respectively. The software packages are available on the GSO Server CD-ROM that is shipped with the product.

Table 24. Software Packages for GSO Servers

Platform	GSO Function	Software Package
AIX	GSO Master or Replica Server	gso.com.rte 2.0 gso.server.dce 2.0
	GSO Database Server	gso.com.rte 2.0 gso.db_odbc.rte 2.0 (ODBC DB Server) gso.db_oci .rte 2.0 (OCI DB Server) gso.db_ctlib.rte 2.0 (CTLIB DB Server)
Solaris	GSO Master or Replica Server	IGSOcom IGSOsrvr
	GSO Database Server	IGSOcom IGSOodbc (ODBC DB server) IGSOoci (OCI DB Server) IGSOctlib (CTLIB DB Server)
Windows NT	GSO Master or Replica Server	\nt\gso\setup.exe
	GSO Database Server	\nt\db\odbc\setup.exe \nt\db\oci\setup.exe

Table 25. Software Packages for GSO Clients

Platform	GSO Function	Software Package
Windows NT	GSO Client	\gso\nt\setup.exe
	GSO DB Client	\gso\db\odbc (ODBC DB client) \gso\db\oci (OCI DB client) \gso\db\ctlib (CTLIB DB Server)

Platform	GSO Function	Software Package
Windows 95	GSO Client	\gso\w95\setup.exe
	GSO DB Client	\gso\db\odbc (ODBC DB client) \gso\db\oci (OCI DB client) \gso\db\ctlib (CTLIB DB Server)
OS/2	GSO Client	\gso\<language>\install

A.1.3 Configuration of GSO Master Server

The standard GSO product installation configures a DCE security and a DCE directory server together with the GSO master server. This section describes how to configure a GSO master on a already existing DCE client, which preferably runs on the same machine as a DCE master or replica server. The GSO master server also runs a DASCOM Directory Service Broker (DSB) and the GSO daemon (gsod).

It is recommended to install the GSO master server on the security master server. This is for performance reasons, because the GSO server uses the DCE security registry heavily to store and retrieve its persistent data, for example the data of the Personal Key Manager (PKM).

The `gsocfg -first` command is used configure a GSO master:

```
gsocfg -first -sradm Acct [-sapwd Pwd] -cadm Acct [-pwd Pwd][-srvrpwd Pwd]
        [-dceum Type] [-dcegrp Grp] [-dceorg Org]
```

The parameters have the following meaning:

`-first` Indicates configuration for the first GSO server in the cell (rather than a replica server).

`-sradm Acct` Specifies a DCE account to become the GSO senior administrator. This account is added to the gso-sr-admin group.

`-sapwd Pwd` Specifies the password for the GSO senior administrator account. When the senior administrator value is not a DCE account, this password is needed, and you are prompted for it if it is not passed.

`-cadm Acct` Specifies the name of the DCE cell administrator's account, which is used to obtain cell administrator credentials.

`-pwd Pwd` Specifies the password of the cell administrator's account. If this flag is not specified with `-first`, you are prompted for the cell administrator password.

`-srvrpwd Pwd` Specifies the password for the GSO server principal, gso_server. With the `-first` flag, this is the password used when the gso_server account is created.

`-dceum Type` Specifies the DCE user management policy where `Type` is replaced by one of the following:

`none`	DCE user management (UM) not allowed in GSO. DCE accounts haves to be added by using the DCE methods.
`sradmin`	DCE UM allowed only by a senior administrator. DCE accounts can be added by the GSO senior administrators who are members of the `gso-sr-admin` group.
`admin`	DCE UM allowed by a senior administrator and a administrators who are members of the `gso-admin` group.

`-dcegrp Grp` Specifies the DCE group to use when creating new DCE accounts through GSO.

`-dceorg Org` Specifies the DCE organization to use when creating through GSO.

Note on Windows NT

The `gsocfg` command is actually a TCL (Tool Command Language) script that needs a TCL interpreter, such as `dcecp`. Windows NT does not support direct calling TCL scripts. The `gsocfg` command needs to be run as follows on Windows NT:

```
cd <GSO_bindir>
dcecp gsocfg.tcl <parameter list>
```

(GSO_bindir is the GSO binary directory, which by default is `C:\ibmgso\bin`. `<parameter list>` is the list of parameters according to the description above and following below.)

The `gsocfg -first` command does all the configuration of DCE and the GSO server. In particular, it carries out the following steps:

1. Checks input parameters and DCE prerequisites
2. Sets up the GSO cell:
   - Creates required DCE principals and the following groups:

     `gso-sr-admin` GSO senior administrators
     `gso-admin` GSO administrators
     `gso-mts`

`gso-user` GSO users

- Creates the GSO server principal `gso_server` with the appropriate access control lists. The server principal is also added to the `acct-admin` group.
- Creates the RPC entries `/.:/gso-servers`, `/.:/gso-keyed-servers` and `/.:/gso-cfg-servers` which hold the RPC information about the GSO service.
- Creates the DCE Extended Registry Schemes and sets the initial values for the `gso_server` principal:

  `/.:/sec/xattrschema/gso_um` GSO user management schema. The corresponding ERA holds information about various aspects of the accounts role within the GSO.

  `/.:/sec/xattrschema/gso_gim` GSO GIM schema

  `/.:/sec/xattrschema/gso_pkm` GSO PKM schema

  `/.:/sec/xattrschema/gso_ptkt` GSO PTKT schema

  `/.:/sec/gso_policy_rgy_mgmt` GSO policy Schema for registry management. Values are: 0 (DCE user management (UM) not allowed in GSO), 1 (DCE UM allowed only by senior administrators), or 2 (DCE UM allowed by senior administrator and administrators).

  `/.:/sec/xattrschema/gso_policy_group` Default group for accounts created by GSO. Entry for server principal contains group name.

  `/.:/sec/xattrschema/gso_policy_org` Default organization for accounts created by GSO. Entry for server principal contains organization name.

  `/.:/sec/xattrschema/gso_policy_tivoli_mgmt` GSO Policy Schema-Tivoli Management. Values are 0 (Tivoli GSO user management disabled) and 1 (Tivoli GSO user management enabled).

3. Sets up the GSO master server:
   - Creates the server-specific DCE RPC entry `/.:/<hostname>/gso-server`.
   - Adds the server to the keytab file.
   - Registers the server with the DCED server process. The GSO server process, `gsod`, will be started automatically when the DCED server process is started, or on request.
   - Adds the server to the configured GSO server group—that is `/.:/gso-cfg-servers`.

- Starts the GSO server to configure the GSO cell. The GSO server master key, /var/gso/etc/.mkey, is created. The server terminates itself.
- Starts the GSO server by the DCED server process.
- Configures the DASCOM Directory Service Broker, dsb. During this process, the intraverse/dsb-servers group and the intraverse/dsb/default/<hostname> principals are created.
- The start of the DASCOM Directory Service Broker, dsb, is added to /etc/inittab.

The gsocfg -first command basically carries out all the necessary configuration steps for the GSO master server.

A.1.4 Configuration of GSO Replica Servers

As soon as there is GSO master server, GSO replica server may be configured. The standard GSO product installation process configures a DCE security and a DCE directory server replica together with the GSO replica. This section describes how to configure a GSO replica on a DCE client, which is not necessary a DCE replica server. The GSO replica will also run a DASCOM Directory Service Broker (DSB).

The GSO replica server may be configured on any DCE client machine; however, it is recommended to install a GSO replica server on a security replica server. This is for performance reasons, because the GSO server uses the DCE security registry to store and retrieve its persistent data. However, in environments with other needs, for example an environment which incorporates wide area network (WAN) links, it may be desirable to install a GSO server without a DCE security replica.

The gsocfg -rep command is used to configure a GSO replica on an existing DCE client:

```
gsocfg -rep -cadm Acct [-pwd Pwd] [-srvrpwd Pwd]
```

The parameters have the following meaning:

-rep	Indicates configuration for GSO replica server.
-cadm Acct	Specifies the name of the DCE cell administrator's account, which is used to obtain cell administrator credentials.
-pwd Pwd	Specifies the password of the cell administrator's account. If this flag is not specified, the gsocfg command will prompt for the cell administrator's password.

`-srvrpwd Pwd` Specifies the password for the GSO server principal, gso_server. If the `-srvrpwd` flag is not used with the `-rep` and `-chgskey` flags, you are prompted for the gso_server account's password.

The tasks of the GSO replica configuration performed by the `gsocfg -rep` command are:

1. Check input parameters and the prerequisites on already available DCE and GSO services.
2. Set up the GSO replica server:
   - Create the server specific DCE RPC entry `/.:/<hostname>/gso-server`.
   - Add the server to the keytab file.
   - Register the server with the `dced` server process. The GSO server process, `gsod`, will then be started automatically when the `dced` server process is started, or on request.
   - Add the new server to the configured GSO server group—that is `/.:/gso-cfg-servers`.
   - Start the new GSO server using the `dced` server process.
   - Configure the DASCOM Directory Service Broker, `dsb`. During this process, the `intraverse/dsb-servers` group and the `intraverse/dsb/default/<hostname>` principals are created.
   - The start of the DASCOM Directory Service Broker, `dsb`, is added to the `/etc/inittab file.`

As shown above, running the `gsocfg -rep` command configures a GSO replica server into an existing DCE cell.

A.1.5 Configuring the Tivoli GSO User Management

Once the GSO master server of the GSO cell is configured, the Tivoli GSO User Management can be configured (Figure 113). This establishes the link between the Tivoli Management Region (TMR) and the GSO cell server.

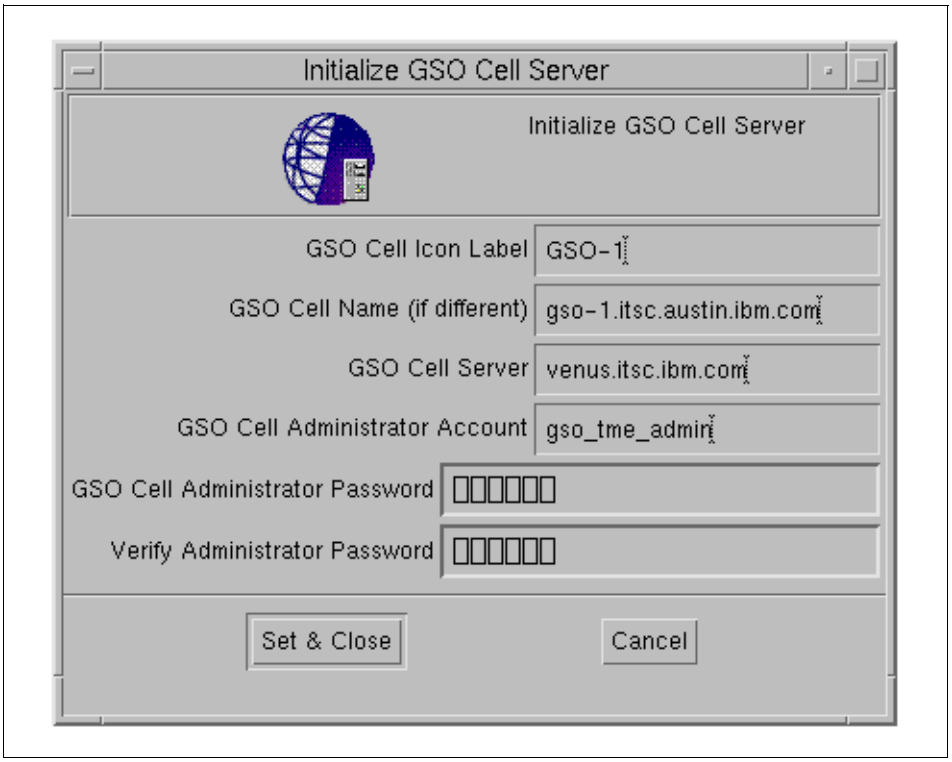


Figure 113. Initializing the GSO Cell Server

The dialog window shown in Figure 113 pops up when **GSOCell...** is selected from the **Create** pull-down menu in a policy region window or, in other words, when you add a new GSO cell to a policy region. (Prior to this, you must have added the GSOCell resource to the list of managed resources within that policy region.)

A.2 Configuration of Additional Directory Service Brokers

The DASCOM Directory Service Broker (DSB) gets configured with every GSO server. However, due to the concepts of DCE, it is possible to run the DSB service on any DCE client that has the DCE security and DCE cell directory (CDS) client configured.

The Directory Service Broker forwards resource location requests from GSO clients to the DCE Cell Directory Service (CDS) and returns this information to the GSO client. The DSB server also caches the cell directory data in a local filesystem cache, `/var/dce/dsb/cds/cds_cache.<version>`. Because of this

caching, and for load balancing and reliability reasons, it may be desirable to configure additional DSB services within the GSO cell.

The following steps describe how to configure additional DSB services on a DCE client:

1. Log in as `root` on UNIX or as `Administrator` on Windows NT.
2. Install the GSO 2.0 Server software package. This software package contains the DASCOM Directory Service Broker.
3. Check that the DCE client has been configured and started and that the security and the directory services are available.
4. Start the DSB service:

   On UNIX:

   ```
   # dsb
   ```

   On Windows NT:

   ```
   cd <Pathname_of_DCE_binaries
   dsb
   ```

5. DSB then asks you for an administrator ID and password in order to configure itself into the cell. It creates the necessary group, principal, and account and creates a keytab file for subsequent authentication with the cell.
6. In order to automatically restart the DSB service at boot time:

   On UNIX:

   Add the following line to `/etc/inittab`, either manually or by using the `mkitab` command (on AIX). The start of the GSO service should be proceeded by the start of the DCE client.

   ```
   gsocfg:2:wait:/usr/bin/gsocfg -start dsb >/dev/console 2>&1
   ```

   On Windows NT:

   The start of the DSB service on the Windows NT system also adds this service as an NT service. Therefore, the DSB service is started automatically at system restart.

The installation of the DSB service is now ready. You may configure your clients to use the additional DSB service.

A.3 Advanced Windows Client Configuration

In order to participate in the GSO cell, the DASCOM NetSEAT lightweight DCE client, as part of the GSO client, has to be configured on a Windows NT or Windows 95 client system.

It is recommended to do the configuration with the TME GSO Plus module, as explained in 5.2, “Installing Clients Using Tivoli Software Distribution” on page 104. This allows an administrator to specify up to three GSO servers to be used by any given client. If the GSO servers have been set up using the standard procedures—that is through the Tivoli GSO Plus module—this procedure is sufficient. The same holds true if the `cfgclient` command is used for client configuration, as described to 5.3, “Native Client Installation and Configuration” on page 118. However, the standard client configuration assumes that each of the GSO servers run a DCE security server, a directory server, a Directory Service Broker, and a GSO server. In order to use servers, which run only a subset of the DCE and GSO services, another client configuration process has to be chosen.

Note

This configuration process uses commands of the underlaying NetSEAT DCE client that are not published in the GSO product documentation. The interfaces may change or disappear with future releases of the product or updates to the product. The information given herein is provided as is.

The following commands configures a client where the different DCE services reside on different nodes:

```
config -c <cellname> [-S secd-host] [-C cdsd-host] [-T dtsd-host] [-D
dsb-host]
```

Use the `-S`, `-T`, `-C` and `-D` options to specify the security, time, CDS, and DSB servers individually. (There is also a `-A` option that allows you to specify a single host that runs all the services.) The location is specified either as an IP address or as a hostname. The `-S`, `-T`, `-C`, `-D` and `-A` options may occur multiple times to specify more than one server. The location for the GSO server daemon (`gsod`) is selected randomly by the client from the list of GSO servers that are registered in the DCE RPC profile, `/.:/gso-servers`.

Tip

The NetSEAT `config` command supports some other options. You can get some usage help by issuing `config -h`. Running this command without any parameters returns the current configuration status. See also A.5.3, “GSO Client Commands” on page 249.

A.4 Recovering from a Failed Server Configuration

There may be situations where the configuration of a GSO server fails and the execution of the Tivoli task *Configure GSO Master Server* subsequently fails with the following error message:

```
This machine is already configured and cannot be configured again
```

To recover from this situation, you may remove the GSO master server including the DCE:

```
cd <GSOBIN directory>
./gsogsoc.tcl gmsu /usr/lpp/gso cell_admin <password_for_cell_admin>
```

Note

Running the `gsogsoc.tcl gmsu` command also unconfigures the underlying DCE cell.

If the server was configured as a replica server, run the following command, which unconfigures also the underlying DCE replica:

```
cd <GSOBIN directory>
./gsogsoc.tcl grsu /usr/lpp/gso cell_admin <password_for_cell_admin>
```

On Windows NT, the GSO installation path needs to be set accordingly. Furthermore, since Windows NT does not support calling TCL scripts directly, you need to precede the `gsogsoc.tcl` command with the `dcecp` command, for example:

```
cd \ibmgso\bin
dcecp gsogsoc.tcl gmsu c:\ibmgso cell_admin <password_for_cell_admin>
```

A.5 Useful Commands

This section lists some commands to list or modify information about GSO components.

Note

Some of these commands are not be supported by the standard product and may change or disappear with product updates.

A.5.1 GSO Server Commands

`gso_err <error_number>` Provides description of error messages from the Tivoli GSO User Administration components. The error_number has the form `0xaabccdd`.

A.5.2 Tivoli/GSO Commands

`wlookup -r GSOCell -a` List s all GSO cells that are defined to Tivoli.

`wgsostartd -n "<cellname>-GSO Cell"` Starts the `tivgsod` process. The process should be started automatically when requests for user management are made.

`wgsostopd -n "<cellname>-GSO Cell"` Stops the `tivgsod` process.

`wgsogetendpt -n "<cellname>-GSO Cell" [ -h | -c | -r | -p | -C ]` Gets information about the GSO endpoint. There are also commands to create, delete, change, and move endpoints: `wgsocrendpt`, `wgsodelendpt`, `wgsochendpt` and `wgsomvendpt`.

A.5.3 GSO Client Commands

The GSO client commands reside in same directory as all GSO binaries, by default under `C:\IBMGSO\BIN`. A very useful command is the NetSEAT/DCE configuration command `config` (see also A.3, “Advanced Windows Client Configuration” on page 247). When called with the `-h` parameter, it gives you the following usage instructions:

```
Usage:
config -h
config -p path
config -d default-cell
config -m compatibility-mode
config -t max-time-delta
config -c cell [-S secd-host] [-C cdsd-host] [-T dtsd-host] [-D dsb-host]
config -c cell [-A host]
Use the -h option to display this message.
Use the -p option to specify the location of the NetSEAT installation.
Use the -d option to specify the name of the default cell. For example:
cell.domain.name
```

```
Use the -m option to toggle the availablility of a compatibility mode. The
compatibility mode is one of 'digital' or 'gradient'.
Use the -t option to specify the maximum difference between host time and
cell time in minutes. For example: 15
Use the -c option to specify the server locations for a particular cell.
For example: cell.domain.name. This options must be accompanied with either
one or more of the -S, -T, -C, or -D options or the -A option
Use the -S, -T, -C and -D options to specify security, time, CDS and DSB
server locations when they are not on the same machine. The location is
specified either as an IP address or hostname
Use the -A option to specify the location of these servers when they are all
on the same machine.
The -S, -T, -C, -D and -A options may occur multiply.
```

The `config` command may be useful when you want the GSO client to connect to specific servers or to review the current configuration by calling it without any parameters.

A.6 GSO-Specific Extended Registry Attributes

The xattrschema objects define schemata, essentially data type information and storage characteristics, for ERAs stored in the DCE Security Registry. GSO defines a schema for each data class it uses. Though several schemata are of the same actual type, assigning different names to them allows easy selection among the data classes (for example, user state information is needed at a different time than target logon information).

Note

While it is best to think of the GSO ERAs as BLOBs, the data structures stored in them are provided here. The structures are subject to change.

All binary data is stored in big endian format.

A data structure used throughout the remaining definitions is `sso_str_t`. Its IDL definition is:

```
typedef struct _sso_str_t {
    unsigned                long  codePage;
    unsigned                long  stringSize;
    [string]                char  locale[LOCALE_STRING_LENGTH];
    [ptr,max_is(stringSize)] char *string;
  } sso_str_t;
```

Following is a list of ERAs provided for your information.

`/.:/sec/xattrschema/gso_um` Byte encoding, attached to user's principal, stores user state data.

```
sso_str_t            annotation;  /* User description */
um_user_acct_state_t  acct_state;  /* User account state */

/* acct_state constants */

const signed32 ACCT_ACTIVE   = 1;
const signed32 ACCT_INACTIVE = 2;
const signed32 ACCT_UNKNOWN  = 3;
```

`/.:/sec/xattrschema/gso_gim` UNUSED, byte encoding, attached to server principal, stores global state data.

```
signed32 gim_chg_pwd_mode_t;
```

`/.:/sec/xattrschema/gso_pkm` Byte encoding, attached to user's principal; each instance stores a target definition, stored keys are encrypted.

Structure:

```
sso_str_t           user_tgt_name;  /* User target name--the string */
                                    /* label assigned to the target */
sso_str_t           tgt_type;       /* Target's type--actual string
                                    /* typename; e.g., WINDOWS_NT40 */
sso_str_t           domain_name;    /* Target's domain name */
sso_str_t           host_name;      /* Target's host name */
sso_str_t           appl_name;      /* Target's applic. name */
sso_str_t           tgt_user_name;  /* User name on target; AKA target
userid */
pkm_user_tgt_type_t  user_tgt_type;  /* User target type--used to */
                                    /* distinguish stored passwords */
                                    /* from one-time passwords */
pkm_key_info_t      key_info;       /* Key information; */
                                    /* the key field will be NULL */
                                    /* for 'without_key' query */
pkm_user_pref_t     user_pref;      /* User configuration preference */
sso_str_t           pref_tgt_name;  /* The preferred program */
sso_str_t           prereq_tgt_name;/* Prerequisite target name */
sso_str_t           ptkt_appl_name; /* Passticket application name */

/* user_tgt_type constants: */
const signed32 USERTGT_PASSWORD   = 1;  /* Persistent user password */
const signed32 USERTGT_PASSTICKET = 2;  /* RACF one-time password */

/* pkm_key_info_t: */
typedef struct pkm_key_info {
  pkm_pwd_policy_t pwd_policy;  /* Password policy, unused in rel 1 */
```

```
  pkm_key_type_t   key_type;    /* Key type */
  sso_str_t        key;         /* Key itself */
} pkm_key_info_t;

/* pwd_policy constants: */
const signed32 PWDPOLICY_PASSTICKET   = 1;
const signed32 PWDPOLICY_TEMPORARY    = 2;
const signed32 PWDPOLICY_MAPPED       = 3;
const signed32 PWDPOLICY_KNOWN        = 4;
const signed32 PWDPOLICY_SYNCHRONIZED = 5;

/* key_type constants: */
const signed32 KEY_SECRET = 1;  /* Secret key based password */
const signed32 KEY_PUBLIC = 2;  /* Public key based password */

/* pkm_user_pref_t: */
typedef struct pkm_user_pref {
  flag           regularly_used; /* Reg. used target, unused in rel 1 */
  pkm_logon_pref_t  logon_pref;      /* Target logon preference */
  pkm_logoff_pref_t logoff_pref;     /* Target logoff preference */
} pkm_user_pref_t;

/* logon_pref constants: */
const signed32 LOGONPREF_NO        = 1;
const signed32 LOGONPREF_IMMEDIATE = 2;
const signed32 LOGONPREF_POSTPONE  = 3;
const signed32 LOGONPREF_SEPARATE  = 4;

/* logoff_pref constants: */
const signed32 LOGOFFPREF_FORCE    = 1;
const signed32 LOGOFFPREF_GRACEFUL = 2;
const signed32 LOGOFFPREF_DISALLOW = 3;
```

`/.:/sec/xattrschema/gso_ptkt` Byte encoding, attached to server principal; each instance stores an encrypted passticket secret key.

```
sso_str_t            appl_server_name; /* Target's appl. server name */
sso_str_t            key;              /* Secret key for ptkt gen. */
gim_ptkt_alg_type_t  alg_type;         /* Ptkt gen. algorithm type */

/* alg_type constants: */

const signed32 PTKT_PRE_1994_ALG  = 1;
const signed32 PTKT_CURR_ALG      = 2;
```

`/.:/sec/xattrschema/gso_policy_rgy_mgmt` Integer encoding, attached to server principal, stores GSO authorization level required to manage a GSO user in the DCE Security Registry.

```
signed32  rgy_user_mgmt;  /* Min. group membership required to manage */
                            /* a GSO user's DCE account                */
```

rgy_user_mgmt constants:

```
const signed32 SSO_DISALLOWED = 0;
const signed32 SSO_SR_ADMIN   = 1;
const signed32 SSO_ADMIN      = 2;
```

`/.:/sec/xattrschema/gso_policy_tivoli_mgmt` Integer encoding, attached to server principal, stores value indicating whether Tivoli or native administration is active.

```
signed32  tivoli_mgmt;  /* 1 == Tivoli mgmt is active, 0 == native */
```

`/.:/sec/xattrschema/gso_policy_group` Binary encoding, attached to server principal, stores DCE group value for GSO-created DCE accounts; config defaults it to the value "none".

```
sec_rgy_name_t  dce_group;  /* Group for GSO-created accounts */
```

`/.:/sec/xattrschema/gso_policy_org` Binary encoding, attached to server principal, stores DCE organization value for GSO-created DCE accounts; config defaults it to the value "none".

Structure:

```
sec_rgy_name_t  dce_org;  /* Org for GSO-created accounts */
```

Appendix B. Program Template Files

This appendix lists the Program Template Files (PTFs) shipped with GSO 2.0 and then provides two samples for your reference. Finally, a sample customized PTF is provided.

> **Note**
>
> The listings in this appendix are slightly modified and reformatted for better presentation and readability. Their relevant content is unchanged.

B.1 Supplied Program Template Files

The following lists two of the supplied program template files provided with IBM Global Sign-On for Multiplatforms, Version 2.0 at the time of writing. They are provided for information only. You should consult your version of the product for the latest PTFs and any information they may contain.

Not all templates are provided for all client platforms. The following lists show the PTFs available for each client. PTFs can be found in the `ibmgso\template` directory.

Program Templates available on Windows NT Clients:

- Attachmate EXTRA! 3270 6.2 for Windows NT
- IBM Personal Communications 3270 4.1 for Windows 95/NT
- Wall Data RUMBA 3270 5.x for Windows 95/NT
- Attachmate EXTRA! 5250 6.2 for Windows NT
- IBM Personal Communications 5250 4.1 for Windows 95/NT
- IBM AS/400 Client Access for Windows 95/NT
- Novell NetWare Server (Bindery) for Windows NT
- Novell NetWare NDS for Windows NT
- Lotus Notes 4.X
- Windows NT 4.0
- SnareWorks V2.0

Program Templates available on Windows 95:

- Attachmate EXTRA! 3270 6.2 for Windows 95
- IBM Personal Communications 3270 4.1 for Windows 95/NT
- Wall Data RUMBA 3270 5.x for Windows 95/NT
- Attachmate EXTRA! 5250 6.2 for Windows 95
- IBM Personal Communications 5250 4.1 for Windows 95/NT

- IBM AS/400 Client Access for Windows 95/NT
- IBM LAN Requester manage password in a domain
- Novell NetWare Server (Bindery) for Windows 95
- Novell NetWare NDS for Windows 95
- Lotus Notes 4.X
- Windows NT 4.0
- SnareWorks V2.0

Program Templates available on OS/2 Warp:

- IBM Personal Communications 3270 4.1 for OS2
- IBM LAN Requester logon with domain verification
- IBM LAN Requester logon with local verification
- IBM LAN Requester manage password in a domain
- IBM LAN Requester manage password on a server
- Novell NetWare Server (Bindery) for OS/2
- Novell NetWare NDS for OS/2
- Lotus Notes 4.X

B.2 Sample PTF: template.ptf

This blank PTF is referenced in the *Programmers Guide* as being supplied as an aid to coding your own target definitions. Unfortunately, it has not been included with all versions of the clients in the initial shipments of the product. It is provided here for your reference.

```
[MAIN]
;
;  default_program_name, required, must be enclosed in quotes. GSO uses
;  it as the default for program name in the Add Program user interface.
;
default_program_name = "example"
;
;  format_version, required, should not be changed. GSO uses it to determine
;  which version of GSO you used to create this program template. If you
;  need to specify your own versions, use comments to do so.
;
format_version = 1.0
;
;  target_type, required, is case sensitive. target_type must match
;  the [TARGET_TYPE] defined in a schema file. Use a predefined GSO
;  target type or create a new one. To create a new one, specify it
;  with no spaces and no quotes to guarantee uniqueness:
;
;      type@DNS-name
;
target_type =
;
```

```
;
[SETTINGS]
;
;  logon_sequence, required, defines the sequence in which the program
;  expects start and logon operations to occur. Valid values are:
;
;  prompt             The program requires the start and logon to be
;                     performed in the same operation. The START section
;                     is ignored.
;
;  start_required     The program must be started before logon can occur.
;                     The START section is required.
;
;  no_start_required  The program can be logged on without being
;                     started first and can be started without
;                     subsequently logging on. The START section is
;                     optional.
;
logon_sequence =
;
;
;  change_password_sequence, required, defines the sequence in which the
;  program expects the logon and change password operations to occur. Valid
;  values are:
;
;  logon_required     The program requires a logon before the password can
;                     be changed. Be aware of the following:
;
;                     - If the target is not logged on and
;                       change_password_sequence=Logon_required, the
;                       LOGON_AND_CHANGE_PASSWORD interface string is used
;                       to log on to the target and change the password.
;
;                     - If the target is not logged on but
;                       change_password_sequence is not set to Logon_required,
;                       the LOGON interface string is used to log on to the
;                       target, and then the CHANGE_PASSWORD interface string
;                       is used to change the password.
;
;                    - If the target is already logged on, the CHANGE_PASSWORD
;                       interface string is used to change the password.
;
;  no_logon_required  The program allows a password to be changed whether
;                     the user is logged on or not. The CHANGE_PASSWORD
;                     interface string is always used to change the password.
;
change_password_sequence =
;
;
;  You must define either minimum_timeout or maximum_timeout. They both
;  default to 0, which implies an infinite wait. If you define both,
;  define minimum_timeout to be less than maximum_timeout.
;
;  minimum_timeout is the minimum amount of time, in seconds,
```

```
;  that GSO should wait for a function to complete before returning.
;  minimum_timeout applies to command line interfaces only. It is
;  intended to be used for targets that return successfully right away
;  but require initialization time before a subsequent operation,
;  such as LOGON, can be performed.
;
;  maximum_timeout is the maximum amount of time, in seconds,
;  that GSO should wait for a function to complete before returning.
;  maximum_timeout applies to both API and command line interfaces.
;  It is intended to prevent a hang situation when a running process
;  does not return when expected.
;
minimum_timeout =
maximum_timeout =
;
;
;  directory, optional, defines the directory containing the files
;  specified in the interface string definitions.  In most cases, this is
;  the default installation directory.  If you do not specify a default
;  directory here, you must specify the fully-qualified file names in the
;  interface string definitions.  You can override the directory specified
;  here by specifying the fully-qualified file names in the interface string
;  definition.
;
directory =
;
;
;  retries, optional, defines the number of times to retry an operation
;  before returning an error. Retries are attempted only when a return code
;  defined in the rc_error return code bucket is received.
;
retries =
;
;
; The Interface Sections
;
;  Interface sections define the interfaces GSO will use to invoke your
;  program to access the particular target type defined in the [SETTINGS]
;  section.  In general, the interfaces define either an API call or a
;  command line interface and the parameters expected.  You can use
;  substitution variables in the parameters to represent target- and
;  program-specific information needed to invoke the function.
;
;  Substitution Variables
;
;   Substitution variables are reserved variables for target-specific
;   information. The following substitution variables are commonly
;   used for all target types:
;
;   $U   The target userid.
;   $P   The target password.
;   $N   A user's new target password. GSO uses this value only used in
;        the CHANGE_PASSWORD and LOGON_AND_CHANGE_PASSWORD interfaces.
;
```

```
;   The following substitution variables are used in some combination
;   to identify the target system. Their exact meaning is defined in the
;   schema file for the particular target type supported:
;
;   $A   The target application name.
;   $D   The target domain name.
;   $H   The target host name.
;
;   $M, a reserved substitution variable, is a special message output
;   parameter that lets you specify a text message that can be returned by
;   the interface. GSO will write this message to the GSO log. Use this
;   parameter when you develop wrapper code to improve the integration of
;   your program with GSO. Use it to give the user information should your
;   interface not complete successfully when invoked by GSO.
;
;  Unreserved Program-Unique Variables
;
;   There are seven other substitution variables--$1 through $7--that allow
;   you to specify other program-specific information. If you use any of
;   these variables, include a section in the program template file to
;   define them.
;
;
;   The capability keyword describes the type of programming interface
;   required to start the program supported by this program template.
;   Valid values are:
;
;   API32    For 32-bit APIs.
;   CLI      For command line.
;
;   For example, capability = API32
;
;   The interface string defines how to invoke the API or command line
;   interface for this function.
;
;  Specifying an API Interface
;
;   interface = "<path-filename combo> function_return_type
function_name(parm1,parm2,...)"
;
;   Specify the interface string, which cannot be longer than 1024 characters,
;   on one line; there are no continuation characters. Enclose the string in
;   double quotes. Enclose the entire executable path and file name
;   combination (path-filename combo) in less than (<) and greater than (>)
;   symbols. The interface is _System linkage.
;
;   Here is a description of the interface parameters:
;
;   function_return_type=int, uint, long, ulong, short, or ushort
;
;   function_name is the API entry point.
;
;   parm1, parm2, and so on are parameters specified as:
;
```

```
;     [parm_direction] parm_type parm_value
;
;     [parm_direction] = [in]
;                        [out,max_is(size)]
;
;     size is the number of bytes required to hold the output data from the
;     function, such as $M in the example that follows. Note: For release
;     1.0, $M is the only output parameter supported.
;
;     parm_type = int, int*, uint, uint*, short, short*, ushort, ushort*,
;                 long, long*, ulong, ulong*, char, char*, uchar, or uchar*
;
;     parm_value = value
;
;     value is either a "value containing spaces enclosed in double quotes"
;     or a substitution variable.
;
;
;  API Interface Example:
;
;   capability = API32
;   interface = "<c:\appl\logon_ops.dll> int appLogon([in]char*
stringval,[in]ushort $2,[in]char* $D,[out,max_is(512)]char* $M)"
;
;
;  Specifying a Command Line Interface
;
;   capability = CLI
;   interface = "<path-filename combo> parm1_value parm2_value ..."
;
;   Enclose the entire interface string in double quotes. Enclose the entire
;   executable path and file name combination (path-filename combo) in the
;   less than (<) and greater than (>) symbols.
;
;   parm1_value=value
;   parm2_value=value
;
;   value is either a "value containing spaces enclosed in double quotes" or
;   a substitution variable.
;
;  CLI Interface Example
;
;   capability = CLI
;   interface = "<c:\ibmlan\netprog\net2.exe> value $3"
;
;
;  Return Codes
;
;   Identify the category into which the return codes fall. Specify return
;   codes as:
;
;   - A list of return codes separated by commas
;   - A range of return code values
;   - A combination of both
```

```
;
;   rc_success              The operation performed on the target was
;                           successful.
;
;   rc_information          The operation performed on the target was
;                           successful and the target returns information
;                           in the form of a null-terminated character
;                           string.  GSO logs this character string in the
;                           default error log.
;
;   rc_chgpwd_error         A change password operation is necessary. GSO
;                           notifies the user.
;
;   rc_credentials_expired  The user's credentials to the target have
;                           expired.  Certain types of targets have time
;                           limited credentials.  If the credentials have
;                           expired, the user might have to reissue a logon
;                           to that target.
;
;   rc_error                The operation performed on the target produced
;                           an error, but the operation is worth retrying.
;                           GSO retries the operation until it reaches the
;                           maximum number of retries specified when the
;                           program was added.  GSO displays status messages
;                           to the user while it retries the operation and
;                           at the point when it reaches the maximum limit
;                           of specified retries.
;
;   rc_severe_error         The operation performed on the target produced
;                           an error so severe that there is no recovery.
;                           GSO does not retry the operation.
;
;
[START]
;
;
; [START] defines the interface to start the program. If the program does
; not support start, omit this section.
;
capability =
interface =
rc_success =
rc_information =
rc_chgpwd_error =
rc_credentials_expired =
rc_error =
rc_severe_error =
;
;
[LOGON]
;
;
; [LOGON] defines the interface to log on to the target. If the program
; does not support logon, omit this section.
```

```
;
capability =
interface =
rc_success =
rc_information =
rc_chgpwd_error =
rc_credentials_expired =
rc_error =
rc_severe_error =
;
;
[CHANGE_PASSWORD]
;
;
; [CHANGE_PASSWORD] defines the interface to change the user's target
; password.  If the program does not support change password, omit this
; section.
;
capability =
interface =
rc_success =
rc_information =
rc_chgpwd_error =
rc_credentials_expired =
rc_error =
rc_severe_error =
;
;
[LOGON_AND_CHANGE_PASSWORD]
;
;
; [LOGON_AND_CHANGE_PASSWORD] defines the interface to log on to the target
; and change the user's password.  If the program does not support logon
; with change password, omit this section.
;
capability =
interface =
rc_success =
rc_information =
rc_chgpwd_error =
rc_credentials_expired =
rc_error =
rc_severe_error =
;
;
[LOGOFF_FORCE]
;
;
; [LOGOFF_FORCE] defines the interface to force logging off of the target.
; If the program does not support logoff force, omit this section.
;
capability =
interface =
rc_success =
```

```
rc_information =
rc_chgpwd_error =
rc_credentials_expired =
rc_error =
rc_severe_error =


[LOGOFF_GRACEFUL]
;
;
; [LOGOFF_GRACEFUL] defines the interface to log off gracefully (without
; loss of data) from the target. If the program does not support logoff
; graceful, omit this section.
;
capability =
interface =
rc_success =
rc_information =
rc_chgpwd_error =
rc_credentials_expired =
rc_error =
rc_severe_error =
;
;
; Substitution Variables
;
;  Use the unreserved substitution variables, $1 through $7, in the
;  interface sections to represent program-specific information. Then,
;  include a section, [$1], [$2], on so on, to define each one used.
;
;[$1]
;value=how is this specified?
;label="text label" or ???
;help="text string" or ???
;
;  Define a value for each substitution variable using the value keyword.
;  By using substitution variables, you simplify your interface sections.
;
;  You can also use substitution variables for information you need
;  from the user. In this case, you omit the value keyword or use it to
;  specify a default value. Then, define values for the label and help
;  keywords. GSO uses these to build a user interface (entry fields) for
;  collecting the information from the user.
;
;  GSO stores the value of the substitution variables, either specified or
;  collected from the user, in the program information database on the
;  client machine.
;
;
[$1]
;
value =
label =
help =
```

```
;
;
[$2]
;
value =
label =
help =
;
;
[$3]
;
value =
label =
help =
;
;
[$4]
;
value =
label =
help =
;
;
[$5]
;
value =
label =
help =
;
;
[$6]
;
value =
label =
help =
;
;
[$7]
;
value =
label =
help =
```

B.3 Sample PTF for Attachmate EXTRA!

Attachmate EXTRA! PTFs are supplied for 3270 and 5250 communications on Windows NT and Windows 95. The only difference between the PTFs is the default program name and the target type. The PTF printed here is the `ext52wnt.ptf`, as found in the `ibmgso\template\nt40\attachmt\extra` directory.

```
[MAIN]
default_program_name = "Attachmate EXTRA! 3270 6.2 for Windows NT"
```

```
format_version = 1.0
target_type = "3270_EMULATION"

[SETTINGS]
logon_sequence = prompt
change_password_sequence = logon_required
minimum_timeout = 0
maximum_timeout = 180
directory = "%IBMGSOPATH%\"

[LOGON]
capability = API32
interface = "<bin\gso3ext.dll> ulong emLogon([out,max_is(512)]char*
$M,[in]char* $U,[in]char* $P,[in]char* $A,[in]char* $H,[in]char* $2,[in]char*
$1,[in]char* $3,[in]char* ehlapi32)"
;
;                ULONG emLogon(
;                                [OUT] Message_String,
;                                [IN]  Userid,
;                                [IN]  Passtkt/Password,
;                                [IN]  Application_Name,
;                                [IN]  System_Name,
;                                [IN]  Session_Id,
;                                [IN]  Session_profile_name,
;                                [IN]  Logon_Script_Filename,
;                                [IN]  EHLLAPI_Dll_name )
;
;           GSO3EXT.DLL is a Dynamic Link Library provided by IBM GSO to
;           perform EHLLAPI functions which accomplish the 3270 target
;           actions
;
;           emLogon is an entry point within GSO3EXT.DLL which
;           accomplishes the logon action
;
;           $M represents the message character string returned for GSO to
;           log to the GSO error log.  This message is intended to relay
;           general error information in the event an unsuccessful return
;           code is received from 'emLogon'
;
;           $U represents the 3270 host system userid
;
;           $P represents the 3270 host system passticket or password
;           for the user specified in $U
;
;           $H represents the host system.  That is, the name of the
;           host system (e.g., TSO2, AUSVM4, etc.) being logged on to.
;
;           $A represents an additional miscellaneous field.  Possible
;           uses are, for example, the name of a 3270 host application
;           to enter after successfully logging on to the host system, or
;           a command to access a gateway menu prior to logging on.
;
;           $2 represents the Session Id associated with the EXTRA! 3270
;           session profile name.  This value is a single letter, A - Z.
```

```
;
;           NOTE: The association between a session profile filename and
;                 a session Id, or short name, is a manual configuration
;                 step for EXTRA! 3270 emulation.
;
;                 To associate a session Id with a session profile filename:
;
;                 - Start a 3270 session using the session profile intended
;                   to be used by GSO for logon
;                 - Select "Options" on the EXTRA! session task bar
;                 - Select "Global Preferences" on the pull down menu
;                 - In the "Advanced" folder, select the desired
;                   session Id and find the session profile name to
;                   associate it with.
;
;           $1 represents the EXTRA! session profile name to launch.  This
;           session profile must have previously been associated with the
;           3270 session Id value specified by $2.
;
;           $3 represents the full path to the .LSF file.  This file
;           defines all of the interaction required to logon to
;           3270 host system.  This file may contain the interaction
;           for all actions supported.  For example, tsosampl.lsf.
;
rc_success = 0
rc_information = 100
rc_chgpwd_error = 99
rc_severe_error = 1-98

[CHANGE_PASSWORD]
capability = API32
interface = "<bin\gso3ext.dll> ulong emChangePW([out,max_is(512)]char*
$M,[in]char* $U,[in]char* $P,[in]char* $N,[in]char* $A,[in]char* $H,[in]char*
$2,[in]char* $3,[in]char* ehlapi32)"
;
;                ULONG emChangePW(
;                                [OUT] Message_String,
;                                [IN]  Userid,
;                                [IN]  Password,
;                                [IN]  NewPassword,
;                                [IN]  Application_Name,
;                                [IN]  System_Name,
;                                [IN]  Session_Id,
;                                [IN]  Logon_Script_Filename,
;                                [IN]  EHLLAPI_Dll_name )
;
;           GSO3EXT.DLL is a Dynamic Link Library provided by IBM GSO to
;           perform EHLLAPI functions which accomplish the 3270 target
;           actions
;
;           emChangePW is an entry point within GSO3EXT.DLL which
;           accomplishes the change password action
;
;           $M represents the message character string returned for GSO to
```

```
;            log to the GSO error log.  This message is intended to relay
;            general error information in the event an unsuccessful return
;            code is received from 'emChangePW'
;
;            $U represents the 3270 host system userid logged on
;
;            $P represents the 3270 host system password for the user
;            specified in $U
;
;            $N represents the 3270 host system new password
;            for the user specified in $U
;
;            $H represents the host system.  That is, the name of the
;            host system (e.g., TSO2, AUSVM4, etc.) being logged on to.
;
;            $A represents an additional miscellaneous field.  Possible
;            uses are, for example, the name of a 3270 host application
;            to enter after successfully logging on to the host system, or
;            a command to access a gateway menu prior to logging on.
;
;            $2 represents the Session Id associated with the 3270 session
;            profile name.  This value is a single letter, A - Z.
;
;            NOTE: See the discussion about session profile name and session
;                  Id under $2 (session Id) in the [LOGON] section above
;
;            $3 represents the full path to the .LSF file.  This file
;            defines all of the interaction required to change the userid's
;            password on the 3270 host system.  This file may contain the
;            interaction for all actions supported.  For example,
;            tsosampl.lsf.
;
rc_success = 0
rc_information = 100
rc_chgpwd_error = 99
rc_severe_error = 1-98


[LOGON_AND_CHANGE_PASSWORD]
capability = API32
interface = "<bin\gso3ext.dll> ulong emLogonC([out,max_is(512)]char*
$M,[in]char* $U,[in]char* $P,[in]char* $N,[in]char* $A,[in]char* $H,[in]char*
$2,[in]char* $1,[in]char* $3,[in]char* ehlapi32)"
;
;                ULONG emLogonC(
;                                [OUT] Message_String,
;                                [IN]  Userid,
;                                [IN]  Password,
;                                [IN]  NewPassword,
;                                [IN]  Application_Name,
;                                [IN]  System_Name,
;                                [IN]  Session_Id,
;                                [IN]  Session_profile_name,
;                                [IN]  Logon_Script_Filename,
```

```
;                                [IN]  EHLLAPI_Dll_name )
;
;           GSO3EXT.DLL is a Dynamic Link Library provided by IBM GSO to
;           perform EHLLAPI functions which accomplish the 3270 target
;           actions
;
;           emLogonC is an entry point within GSO3EXT.DLL which
;           accomplishes the logon and change password action
;
;           $M represents the message character string returned for GSO to
;           log to the GSO error log.  This message is intended to relay
;           general error information in the event an unsuccessful return
;           code is received from 'emLogonC'
;
;           $U represents the 3270 host system userid
;
;           $P represents the 3270 host system password for the user
;           specified in $U
;
;           $N represents the 3270 host system new password
;           for the user specified in $U
;
;           $H represents the host system.  That is, the name of the
;           host system (e.g., TSO2, AUSVM4, etc.) being logged on to.
;
;           $A represents an additional miscellaneous field.  Possible
;           uses are, for example, the name of a 3270 host application
;           to enter after successfully logging on to the host system, or
;           a command to access a gateway menu prior to logging on.
;
;           $2 represents the Session Id associated with the 3270 session
;           profile name.  This value is a single letter, A - Z.
;
;           NOTE: See the discussion about session profile name and session
;                 Id under $2 (session Id) in the [LOGON] section above
;
;           $1 represents the EXTRA! session profile name to launch.  This
;           session profile must have previously been associated with the
;           3270 session Id value specified by $2.
;
;           $3 represents the full path to the .LSF file.  This file
;           defines all of the interaction required to logon to the
;           3270 host system and change the password. This file may
;           contain the interaction for all actions supported.  For
;           example, tsosampl.lsf.
;
rc_success = 0
rc_information = 100
rc_chgpwd_error = 99
rc_severe_error = 1-98

[LOGOFF_GRACEFUL]
capability = API32
```

```
interface = "<bin\gso3ext.dll> ulong emLogoff([out,max_is(512)]char*
$M,[in]char* $2,[in]char* $3,[in]char* ehlapi32)"
;
;                ULONG emLogoff(
;                                [OUT] Message_String,
;                                [IN]  Session_Id,
;                                [IN]  Logon_Script_Filename,
;                                [IN]  EHLLAPI_Dll_name )
;
;           GSO3EXT.DLL is a Dynamic Link Library provided by IBM GSO to
;           perform EHLLAPI functions which accomplish the 3270 target
;           actions
;
;           emLogoff is an entry point within GSO3EXT.DLL which
;           accomplishes the graceful logoff
;
;           $M represents the message character string returned for GSO to
;           log to the GSO error log.  This message is intended to relay
;           general error information in the event an unsuccessful return
;           code is received from 'emLogoff'
;
;           $2 represents the Session Id associated with the 3270 session
;           profile name.  This value is a single letter, A - Z.
;
;           NOTE: See the discussion about session profile name and session
;                 Id under $2 (session Id) in the [LOGON] section above
;
;           $3 represents the full path to the .LSF file.  This file
;           defines all of the interaction required to logoff. This file
;           may contain the interaction for all actions supported.
;           For example, tsosampl.lsf.
;
rc_success = 0
rc_information = 100
rc_severe_error = 1-99

[$1]
value = "c:\extra\sessions\tso1.edp"
label = "Session Profile"
help = "A session profile; the file EXTRA! uses to store the configuration
information regarding a session."

[$2]
value = "A"
label = "Session ID"
help = "The Session ID is the short session name associated with the 3270
session.  EHLLAPI will not work correctly unless this association has been
made."

[$3]
value = "%IBMGSOPATH%\script\3270_em\tsosampl.lsf"
label = "Script File Path"
```

```
help = "The full path of the script file (.LSF).  The file defines the
interaction required to perform an action (e.g., LOGON, LOGOFF, CHANGE
PASSWORD) regarding a host system."
```

B.4 Sample Customized PTF

This is an example of coding your own Program Template File.

```
[MAIN]

default_program_name = "Made Up Version 1"

format_version = 1.0

target_type = "MADEUP_V1"


[SETTINGS]


logon_sequence = No_start_required

change_password_sequence = No_logon_required

minimum_timeout = 0

maximum_timeout = 0

directory="C:\some.path\"


[LOGON]

capability = API32

interface = "<bin\an.dll> ulong an_logon([out,max_is(512)]char* $M, [in]char*
$U, [in]char* $P, [in]char* $A, [in]char* $H [in]char* $D [in]char* $1)"
;
;                     ULONG at_logon(
;                                     [OUT] msg_string,
;                                     [IN]  userid,
;                                     [IN]  password,
;                                     [IN]  database,
;                                     [IN]  host
;                                     [IN]  resource)
;
rc_success = 0
rc_error = 67
rc_severe_error = 1-66,68-3000,9999


[CHANGE_PASSWORD]
; This section defines the interface to change the password on specified
```

```
; host/database
capability = API32
interface = "<bin\an.dll> ulong an_change_pw([out,max_is(512)]char* $M,
[in]char* $U,[in]char* $P,[in]char* $N,[in]char* $A $H)"
;
;                     ULONG at_change_pw(
;                                        [OUT] msg_string,
;                                        [IN]  Userid,
;                                        [IN]  Password,
;                                        [IN]  NewPassword,
;                                        [IN]  database,
;                                        [IN]  host )
rc_success = 0
rc_error = 67
rc_severe_error = 1-66,68-3000,9999


[LOGOFF_GRACEFUL]
; This section defines the interface to gracefully cancel.
capability = API32
interface = "<bin\an.dll> ulong an_logoff([out,max_is(512)]char* $M, [in]char*
$H)"
;
;                     ULONG an_logoff(
;                                        [OUT] msg_string,
;                                        [IN]  host)
;
rc_success = 0
rc_error = 67
rc_severe_error = 1-66,68-3000,9999
[$1]

value = "001"
label = "Table Start Entry"
help = "enter the number of the table row you wish to start at"
```

Appendix C. Schema Files

Printed here is the schema file for the supplied targets available with IBM Global Sign-On for Multiplatforms, Version 2.0. It is provided here for information only, and you should consult the version supplied with your copy of the product for the latest information. Schema files are held on the GSO server in the `/var/gso/schema` directory.

The second schema is an example of coding your own.

Note

The listings in this appendix are slightly modified and reformatted for better presentation and readability. Their relevant content is not changed.

C.1 Supplied Schema File: ibmgso.sch

```
;  (C) COPYRIGHT International Business Machines Corp. 1996,1997
;  All Rights Reserved
;  US Government Users Restricted Rights - Use, duplication or
;  disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
;
; DO NOT MODIFY THIS FILE. YOU WILL LOSE ALL THE CHANGES BETWEEN INSTALLS.
; Please create another file with an sch extension to define information
; about vendor-supplied targets.
;
; This file contains information for all target types provided by IBM.
; The [TARGET_TYPE] here should match the target_type keyword in the
; corresponding program template file.
;
;*************************************************************************
;
; The following is the schema information for a target type of 3270
; emulation.
;
[3270_EMULATION]
;
; target_description describes the target.
;
target_description="3270 emulator"
;
; A= can have one of these values:
;
;     APPLICATION
;     CELL
;     DATABASE
;     DOMAIN
;     HOST
```

```
;     PEER
;     SERVER
;     SESSION
;     SYSTEM
;     WORKSTATION
;     DEVICE
;     RESOURCE
;     USER_CONTEXT
;
A=APPLICATION, OPTIONAL
;
; H = specifies the system.
;
H=SYSTEM
;
; U = specifies the userid.
;
U=USERID
;
; password_capability indicates whether the password capability of the
; target system is PASSTICKET or STORED.
;
password_capability=PASSTICKET
;
; Program_name indicates whether the program is required or optional.
;
Program_name = PROGRAM, REQUIRED
;
;*************************************************************************
;
[5250_EMULATION]
;
; target_description describes the target.
;
target_description="5250 emulator"
;
; H = specifies the system.
;
H=SYSTEM
;
; U = specifies the userid.
;
U=USERID
;
; password_capability indicates whether the password capability of the
; target system is PASSTICKET or STORED.
;
password_capability=STORED
;
; Program_name indicates whether the program is required or optional.
;
Program_name = PROGRAM, REQUIRED
;
;*************************************************************************
;
```

```
;
; This is the schema information for the Windows NT 4.0 server target.
;
; The [TARGET_TYPE] is WINDOWS_NT40.
;
[WINDOWS_NT40]
;
; The target_description is Windows NT 4.0 Server.
;
target_description="Windows NT 4.0 Server"
;
; A, which is DEVICE, indicates the local device to which the connection needs
; to be redirected. If the device is not specified, a connection is made
; without redirecting to a local drive.
;
A = DEVICE, OPTIONAL
;
;H, which is RESOURCE, indicates the remote resource to which the connection
; needs to be established.
;
H = RESOURCE
;
; D is DOMAIN, where the network resource is available.
;
D=DOMAIN
;
; U, which is USERID, indicates that a USERID is required.
;
U=USERID
;
;The password capability is stored.
;
password_capability=STORED
;
;*****************************************************************************
;
;This is the schema information for the Novell Netware Server target.
;
; The [TARGET_TYPE] is NOVELL_NETWARE_SERVER.
;
[NOVELL_NETWARE_SERVER]
;
; The target_description is Novell Netware Server.
;
target_description="Novell Netware Server"
;
; H, which is SERVER, is the remote server to be logged onto.
;
H=SERVER
;
; U, which is USERID, indicates that a USERID is required.
;
U=USERID
;
```

```
;
;The password capability is stored.
;
password_capability=STORED
;
;*************************************************************************
;
;This is the schema information for the Novell Netware NDS target.
;
; The [TARGET_TYPE] is NOVELL_NETWARE_NDS.
;
[NOVELL_NETWARE_NDS]
;
; The target_description is Novell Netware NDS.
;
target_description="Novell Netware NDS"
;
; A, which is USER_CONTEXT, is the NDS context the use uses to logon.
;
A=USER_CONTEXT
;
; U, which is USERID, indicates that a USERID is required.
;
U=USERID
;
;The password capability is stored.
;
password_capability=STORED
;
;*************************************************************************
;
;This is the schema information for the IBM LAN Server domain target
;
;The [TARGET_TYPE] is IBMLS_DOMAIN
;
[IBMLS_DOMAIN]
;
;The target_description is IBM LAN Server logon with domain verification
;
target_description="IBM LAN Server domain logon"
;
;D is the name of the LAN Server domain.
;
D=DOMAIN
;
;U, which is USERID, indicates that a USERID is required.
;
U=USERID
;
;The password capability is stored.
;
password_capability=STORED
;
;*************************************************************************
;
```

```
;
;This is the schema information for the IBM LAN Server local target
;
;The [TARGET_TYPE] is IBMLS_LOCAL
;
[IBMLS_LOCAL]
;
;The target_description is IBM LAN Server logon with local verification
;
target_description="IBM LAN Server local logon"
;
;U, which is USERID, indicates that a USERID is required.
;
U=USERID
;
;The password capability is stored.
;
password_capability=STORED
;
;*****************************************************************************
;
;This is the schema information for the IBM LAN Server manage password target
;
;The [TARGET_TYPE] is IBMLS_MANAGEPWD
;
[IBMLS_MANAGEPWD]
;
;The target_description is IBM LAN Server manage passwords with no
;logon or logoff support
;
target_description="IBM LAN Server manage password on a server"
;
;The variable $A is a LAN Server server name
;This is the server that will manage passwords
;
A=SERVER
;
;U, which is USERID, indicates that a USERID is required.
;
U=USERID
;
;The password capability is stored.
;
password_capability=STORED
;
;*****************************************************************************
;
;This is the schema information for the IBM LAN Server manage password target
;
;The [TARGET_TYPE] is IBMLS_MANAGEPWD_DOM
;
[IBMLS_MANAGEPWD_DOM]
;
;The target_description is IBM LAN Server manage passwords with no
```

```
;logon or logoff support
;
target_description="IBM LAN Server manage password in a domain"
;
;D is the name of the LAN Server domain.
;
D=DOMAIN
;
;U, which is USERID, indicates that a USERID is required.
;
U=USERID
;
;The password capability is stored.
;
password_capability=STORED
;
;*************************************************************************
;
; This is the schema information for the notes 4.X target.
;
; The [TARGET_TYPE] is NOTES_4X
;
[NOTES_4X]
;
; The target_description is LOTUS NOTES 4.X Client.
;
target_description="LOTUS NOTES 4.X Client"
;
;H, which is WORKSTATION, indicates the host name of the GSO client.
;This is optional
;
H = WORKSTATION, OPTIONAL
;
; A, which is APPLICATION, indicates the name of the NOTES id file. This can be
; a fully qualified name or just the file name.
;
A=application
;
; U, which is USERID, indicates that a USERID is required.
;
U=USERID
;
;The password capability is stored.
;
password_capability=STORED
;
;*************************************************************************
;
; The following is the schema information for a target type of IBM AS/400
Client Access
;
[IBM_AS400_CLIENT_ACCESS]
target_description="IBM AS/400 Client Access"
H=SYSTEM
```

```
U=USERID
password_capability=STORED
;
;*****************************************************************************
;
; The following is the schema information for the SnareWorks V2.01 target type
;
;
[SNAREWORKS_201]
target_description="SnareWorks V2.01"
D=CELL
U=USERID
password_capability=STORED
```

C.2 Example Schema File

```
;*****************************************************************************
;
; This is a made-up schema to go with a made-up ptf for a made-up program
;
; The [TARGET_TYPE] is MADEUP_V1
;
[MADEUP_V1]
;
; The target_description is Made Up Version 1'
;
target_description="Made Up Version 1"
;
; A, is going to be labeled database and is optional
;
A=DATABASE, OPTIONAL
;
; H is going to be labelled HOST and is required
;
H=HOST
;
; D is going to be labelled RESOURCE and is also required
;
D=RESOURCE
;
; U, which is USERID, indicates that a USERID is required.
;
U=USERID
;
; The target uses passwords not passtickets so capability is stored.
;
password_capability=STORED
;
;
;*****************************************************************************
```

Appendix D. Logon Script Files

Logon Script Files (LSFs) can be thought of programs that contain the steps to log into a system or application through a terminal emulation program. This appendix lists a sample LSF shipped with the GSO 2.0 product and a sample LSF for logging on to a UNIX system.

Note

The listings in this appendix are slightly modified and reformatted for better presentation and readability. Their relevant content is not changed.

D.1 Supplied Example Logon Script File: tsosampl.lsf

```
;  GSO requires you to supply a logon script file (LSF) so that users can
;  perform 3270 host actions, such as logging on and logging off. An LSF
;  contains the information the client user enters to perform the GSO action.
;  You supply this information through keywords in the LSF.
;
;  Keywords
;
;    Section Keywords
;
;        VERSION:
;        GLOBAL:
;        LOGON:
;        LOGONC:
;        CHANGEPW:
;        LOGOFF:
;        END_FILE:
;
;    Action Keywords
;
;        LOGON:
;        LOGONC:
;        CHANGEPW:
;        LOGOFF:
;
;    Group Keywords
;
;        START:
;        STOP:
;
;    Operation Keywords
;
;        DATA_SEND:
;        LOOK_FOR:
;        LOOK_FOR?
```

```
;        PAUSE:
;        PTKT_SEND:
;
;
;  Section Keywords
;
;    Section keywords are optional except where otherwise noted. GSO does not
;    support duplicate section keyword entries in the LSF. Section keywords
;    are:
;
;    Keyword        Definition
;
;    VERSION: nnnn A required keyword, this is the first line in the LSF.
;                  Do not change this value. The IBM Single Signon Wizard
;                  uses VERSION: to identify the script file format used:
;                  nnnn has a value of 0001 to 9999. To indicate the version
;                  number of a particular host application, use comments.
;
;
;    GLOBAL:       Defines common screens that the user might encounter while
;                  interacting with a host system. Use GLOBAL: to perform
;                  tasks like clearing a screen when a full screen indicator
;                  is displayed, or exiting a host application or menu system
;                  submenu so that the user can successfully log on, log off,
;                  log on and change their password, or change their password.
;
;    LOGON:        This action keyword defines screens that the user might
;                  encounter while logging on to a host system. Follow
;                  LOGON: with one or more groups of screen definitions. Each
;                  screen definition group begins with the START: keyword
;                  and ends with the STOP: keyword.
;
;    LOGONC:       This action keyword defines screens that the user might
;                  encounter while logging on to a host system where a
;                  change password is required or desired. Follow LOGONC: with
;                  one or more groups of screen definitions. Each screen
;                  definition group begins with the START: keyword and ends
;                  with the STOP: keyword.
;
;    CHANGEPW:     This action keyword defines screens that the user might
;                  encounter while changing their password on a host system.
;                  Follow CHANGEPW: with one or more groups of screen
;                  definitions. Each screen definition group begins with
;                  the START: keyword and ends with the STOP: keyword.
;
;    LOGOFF:       This action keyword defines screens that the user might
;                  encounter while logging off of a host system. Follow
;                  LOGOFF: with one or more groups of screen definitions.
;                  Each screen definition group begins with the START:
;                  keyword and ends with the STOP: keyword.
;
;    END_FILE:     This required keyword indicates that input is completed.
;
; Note: Both the GLOBAL: and action (LOGON:, LOGOFF:, and so on) keywords
```

```
; are optional.  However, it is an error to omit both the GLOBAL: and
; the specific action keyword section. For example, if a logon is requested
; and both the GLOBAL: and LOGON: sections are omitted, GSO returns an error.
;
;
;  Group Keywords
;
;    Group keywords are:
;
;    Keyword       Definition
;
;    START:        A search string of one to 255 characters that defines
;                  the start of a screen definition group for a particular
;                  action. The characters that follow this keyword indicate
;                  the specific host screen for which this start sequence is
;                  looking. If GSO finds the string, this screen group begins
;                  the specified actions. If GSO does not find the string,
;                  GSO searches for the next START: keyword character string
;                  inside the action keyword. Use START: inside of an action
;                  section only. Do not nest START: keywords.
;
;    STOP:         Defines the end of a screen definition group for a
;                  particular action.  Each START: keyword requires a closing
;                  STOP: keyword.
;
;
;  Operation Keywords
;
;   Use operation keywords to specify:
;
;     - Strings, such as commands, to send to the host
;     - What host screens to look for
;     - How long to wait before continuing when host timing dictates it
;
;   Place operation keywords in the GLOBAL: or action section. For an
;   action section, place operation keywords within START: and STOP: keywords.
;
;    Operation keywords are:
;
;    Keyword        Definition
;
;    DATA_SEND:    A string of one to 255 characters to be sent to the host
;                  from the current cursor position. Use the enter (@E) and
;                  tab (@T) keys to move to multiple input fields, if
;                  required. If you need to use both enter and tab, enter
;                  them on separate DATA_SEND: lines.
;
;    LOOK_FOR:     A search string of one to 255 characters that indicates
;                  that you want to wait for a specific host screen before
;                  continuing in the sequence.  The string that follows this
;                  keyword determines which host screen to wait for. If a
;                  match is not found, GSO goes into a query/wait loop for
;                  up to 30 seconds to give the screen time to appear. Upon
;                  timeout, GSO returns an error indicating the expected host
```

```
;                 screen was not found.
;
;    LOOK_FOR?    A search string of one to 255 characters that indicates
;                 that you want to take a quick look for a specific host
;                 screen before continuing in the sequence. Unlike
;                 LOOK_FOR:, GSO issues a very short wait and does not
;                 return an error if GSO does not find a match for the
;                 screen. This keyword is useful in the GLOBAL: section
;                 to do a quick check of screen full conditions.
;
;    PAUSE: nn    Indicates the number of seconds GSO is to wait before
;                 processing the next keyword.  This is a conditional wait
;                 in that a host screen update causes the wait to complete.
;                 That is, a wait of the full duration only occurs if the
;                 host screen is not updated during the wait period.
;
;    PTKT_SEND: ********
;                 Represents a passticket to be sent, as keystrokes, to the
;                 host. Follow PTKT_SEND: with eight asterisks. Place any
;                 remaining keystrokes, such as enter or tab, on a
;                 separate DATA_SEND: line.
;
;                 You can also use this keyword to send a password. However,
;                 sending both passtickets and passwords using the &PTKTPW
;                 symbol with the DATA_SEND keyword is more flexible.
;
;
;  Query Operation Keywords
;
;    QUERY_BAD_PW:, QUERY_BAD_USER:, and QUERY_EXP_PW: are query operation
;    keywords intended to detect userid and password failures. If GSO finds a
;    match, GSO ends the action and logs an error message in the GSO error log.
;    If GSO does not find a match, GSO continues processing.
;
;    Query operation keywords cause GSO to take a single "quick look" at the
;    screen to detect the error condition.  Due to host timings, you might need
;    to insert PAUSE: keywords before the query keyword to ensure that the
;    host has responded prior to the "quick look."
;
;    NOTE: You can only place query operation keywords in the action keyword
;    sections where userid and password, and new password values are used
;    (for example, in the LOGON:, LOGON_AND_CHANGE_PASSWORD:, and
;    CHANGE_PASSWORD: sections). Additionally, you must place query operation
;    keywords in a START: and STOP: keyword sequence.
;
;    Query operation keywords are:
;
;    Keyword      Definition
;
;    QUERY_BAD_PW: A search string of one to 255 characters that indicates
;                 that you want to take a quick look for a host application
;                 message indicating the user password or new password has
;                 been rejected for some reason.  It is reasonable that a
;                 host application might have several different messages
```

```
;                that would indicate this state. Therefore, you can insert
;                multiples of this keyword into a sequence.  You can also
;                use the QUERY_BAD_PTKT: keyword instead.
;
;   QUERY_BAD_USER:
;                A search string of one to 255 characters that indicates
;                you want to take a quick look for a host application
;                message indicating the userid is not valid or has been
;                revoked for some reason. It is reasonable that a host
;                application might have several different messages to
;                indicate this. Therefore, you can insert multiples of
;                this keyword into a sequence.
;
;   QUERY_EXP_PW: A search string of one to 255 characters that indicates
;                that you want to take a quick look for a host application
;                message indicating the user password has expired. To GSO,
;                an expired password condition differs from a bad password.
;                Therefore, GSO detects these as separate conditions. You
;                can also use the QUERY_EXPIRED_PW: keyword instead.
;
;
; Reserved Variables
;
;   Use these reserved names as substitution variables for target-defined
;   values:
;
;   Variable      Definition
;
;   &APPL         Is substituted with the application name as defined in
;                 the creation of a target of this target_type. The
;                 application name is the name of the application on the
;                 host system being logged on to.
;
;   &NEWPW        Is substituted with the new password as defined in the
;                 change password operation for this target.
;
;   &PTKTPW       Is substituted with the passticket or password as defined
;                 in the creation of a target of this target_type.
;
;   &SYS          Is substituted with the system name as defined in the
;                 creation of a target of this target_type. The system
;                 name is the name of the host system being logged on to,
;                 for example, TSO2.
;
;   &USERID       Is substituted with the userid as defined in the creation
;                 of a target of this target_type.  The userid is the user
;                 identification of the person logging onto the specified
;                 host system.
;
;
; Reserved Variables
;
;   The most commonly used keyboard mnemonics are:
;
```

```
;
;    @B (Left Tab)    @C (Clear)       @D (Delete)      @E (Enter)
;
;    @F (Erase EOF)   @I (Insert)      @L (Cursor       @N (New Line)
;                                      Left)
;
;    @R (Reset)       @T (Right Tab)   @U (Cursor Up)   @V (Cursor
;                                                          Down)
;
;    @Z (Cursor       @0 (Home)        @1 (PF1)         @2 (PF2)
;        Right)
;
;    @3 (PF3)         @4 (PF4)         @5 (PF5)         @6 (PF6)
;
;    @7 (PF7)         @8 (PF8)         @9 (PF9)         @a (PF10)
;
;    @b (PF11)        @c (PF12)        @d (PF13)        @e (PF14)
;
;    @f (PF15)        @g (PF16)        @h (PF17)        @i (PF18)
;
;    @j (PF19)        @k (PF20)        @l (PF21)        @m (PF22)
;
;    @n (PF23)        @o (PF24)        @x (PA1)         @y (PA2)
;
;    @z (PA3)         @@ (@)           @$ (Alternate    @< (Backspace
;                                          Cursor)          Erase)
;
;    @A@F (Erase      @A@H (System     @A@J (Cursor     @A@Q (Attention)
;          Input)           Request)         Select)
;
;
;    See the EHLLAPI Programming Guide of the emulator you are using for more
;    information on supported keyboard mnemonics.
;
;
; -------- .lsf begin -----------
;
;
VERSION: 0001
;
GLOBAL:
  LOOK_FOR: HOLDING
  DATA_SEND: @C
;
;
LOGON:
  START: DHAT (USIBMTH)
    DATA_SEND: &SYS@E
    PAUSE: 1
    LOOK_FOR: IKJ56700A ENTER USERID
    DATA_SEND: &USERID@E
    PAUSE: 2
    QUERY_BAD_USER: not authorized to use
    LOOK_FOR: Password  ===>
```

```
  DATA_SEND: &PTKTPW@E
  PAUSE: 2
  QUERY_BAD_USER: RACF TEMPORARILY REVOKING USER ACCESS
  QUERY_BAD_PW: PASSWORD NOT AUTHORIZED FOR USERID
  LOOK_FOR: LOGON IN PROGRESS
  LOOK_FOR: OS/390
  DATA_SEND: @C
  PAUSE: 1
  DATA_SEND: @E
  DATA_SEND: @E
STOP:
START: DEVELOPMENT HOST ATTACH TEST
  DATA_SEND: &SYS@E
  PAUSE: 1
  LOOK_FOR: IKJ56700A ENTER USERID
  DATA_SEND: &USERID@E
  PAUSE: 2
  QUERY_BAD_USER: not authorized to use
  LOOK_FOR: Password  ===>
  DATA_SEND: &PTKTPW@E
  PAUSE: 2
  QUERY_BAD_USER: RACF TEMPORARILY REVOKING USER ACCESS
  QUERY_BAD_PW: PASSWORD NOT AUTHORIZED FOR USERID
  PAUSE: 1
  LOOK_FOR: LOGON IN PROGRESS
  LOOK_FOR: OS/390
  DATA_SEND: @C
  PAUSE: 1
  DATA_SEND: @E
  DATA_SEND: @E
STOP:
START: IKJ56700A ENTER USERID -
  DATA_SEND: &USERID@E
  PAUSE: 2
  QUERY_BAD_USER: not authorized to use TSO
  LOOK_FOR: Password  ===>
  DATA_SEND: &PTKTPW@E
  PAUSE: 2
  QUERY_BAD_USER: RACF TEMPORARILY REVOKING USER ACCESS
  QUERY_BAD_PW: PASSWORD NOT AUTHORIZED FOR USERID
  PAUSE: 1
  LOOK_FOR: LOGON IN PROGRESS
  LOOK_FOR: OS/390
  DATA_SEND: @C
  PAUSE: 1
  DATA_SEND: @E
  DATA_SEND: @E
STOP:
START: Password  ===>
  PTKT_SEND: ********
  DATA_SEND: @E
  PAUSE: 2
  QUERY_BAD_PW: PASSWORD NOT AUTHORIZED FOR USERID
  PAUSE: 1
```

```
    LOOK_FOR: LOGON IN PROGRESS
    LOOK_FOR: OS/390
    DATA_SEND: @C
    PAUSE: 1
    DATA_SEND: @E
    DATA_SEND: @E
  STOP:
;
;
LOGOFF:
  START: READY
    DATA_SEND: LOGOFF@E
    LOOK_FOR: 793-6300
  STOP:
  START: Application Status
    DATA_SEND: LOGOFF@E
    LOOK_FOR: 793-6300
  STOP:
  START: ENTER LOGON OR LOGOFF-
    DATA_SEND: LOGOFF
    DATA_SEND: @E
    LOOK_FOR: 793-6300
  STOP:
;
; at ISPF command prompt
;
  START: Option ===>
    DATA_SEND: x
    DATA_SEND: @E
    LOOK_FOR: READY
    DATA_SEND: LOGOFF@E
    LOOK_FOR: 793-6300
  STOP:
;
;
LOGONC:
  START: DHAT (USIBMTH)
    DATA_SEND: &SYS@E
    PAUSE: 1
    LOOK_FOR: IKJ56700A ENTER USERID
    DATA_SEND: &USERID@E
    QUERY_BAD_USER: not authorized to use TSO
    PAUSE: 2
    LOOK_FOR: Password  ===>
    DATA_SEND: &PTKTPW@@@T
    DATA_SEND: &NEWPW@E
    PAUSE: 2
    QUERY_BAD_USER: RACF TEMPORARILY REVOKING USER ACCESS
    QUERY_BAD_PW: PASSWORD NOT AUTHORIZED FOR USERID
    PAUSE: 1
    LOOK_FOR: Reenter the new password in the NEW PASSWORD field for
verification
    DATA_SEND: &NEWPW@E
    LOOK_FOR: LOGON IN PROGRESS
```

```
    LOOK_FOR: OS/390
    DATA_SEND: @C
    PAUSE: 1
    DATA_SEND: @E
    DATA_SEND: @E
  STOP:
;
;
CHANGEPW:
  START: READY
    DATA_SEND: password password@E
    PAUSE: 2
    LOOK_FOR: ENTER CURRENT PASSWORD
    DATA_SEND: &PTKTPW@E
    QUERY_BAD_PW: PASSWORD CHANGE REJECTED
    PAUSE: 2
    LOOK_FOR: ENTER NEW PASSWORD
    DATA_SEND: &NEWPW@E
    QUERY_BAD_PW: PASSWORD CHANGE REJECTED
    PAUSE: 2
    LOOK_FOR: READY
  STOP:
;
; exit from ISPF menu before changing password
;
  START: Option ===>
    DATA_SEND: x
    DATA_SEND: @E
    LOOK_FOR: READY
    DATA_SEND: password password@E
    PAUSE: 2
    LOOK_FOR: ENTER CURRENT PASSWORD
    DATA_SEND: &PTKTPW@E
    PAUSE: 2
    LOOK_FOR: ENTER NEW PASSWORD
    DATA_SEND: &NEWPW@E
    PAUSE: 2
    LOOK_FOR: READY
  STOP:
;
;
END_FILE:
```

D.2 Sample UNIX Logon Script

```
; PCOMM VT Emulation - Telnet to AIX 4.2
;
VERSION: 0001
;
GLOBAL:
;
;
```

```
;
LOGON:
;
;    this START: line is looking for "login:" (spelled exactly) on the screen
  START: login:
    DATA_SEND: &USERID@E
;    this line is looking for "Password:" (spelled exactly) on the screen
    LOOK_FOR: Password:
    DATA_SEND: &PTKTPW@E
    QUERY_BAD_USER: You entered an invalid login name or password
    LOOK_FOR: CMD>
  STOP:
;
LOGOFF:
;
;    this START: line is looking for "CMD>" (spelled exactly) on the screen
  START: CMD>
    DATA_SEND: logout@E
    DATA_SEND: exit@E
  STOP:
END_FILE:
```

Appendix E. Special Notices

This publication is intended to help consultants and technical advisors to understand the operation principles of the IBM IBM Global Sign-On for Multiplatforms, Version 2.0 product. The information in this publication is not intended as the specification of any programming interfaces that are provided by IBM Global Sign-On for Multiplatforms, Version 2.0. See the PUBLICATIONS section of the IBM Programming Announcement for IBM Global Sign-On for Multiplatforms, Version 2.0 for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594 USA.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee

that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

The following document contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples contain the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

AIX ®
CICS
eNetwork
HACMP/6000
OS/2 ®
RS/6000 ®
AS/400 ®
DB2 ®
Global Sign-On
IBM ®
OS/400 ®

The following terms are trademarks of other companies:

Java and HotJava are trademarks of Sun Microsystems, Incorporated.

Microsoft, Windows, Windows NT, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

Pentium, MMX, ProShare, LANDesk, and ActionMedia are trademarks or registered trademarks of Intel Corporation in the U.S. and other countries.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Other company, product, and service names may be trademarks or service marks of others.

Appendix F. Related Publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

F.1 International Technical Support Organization Publications

For information on ordering these ITSO publications, see “How to Get ITSO Redbooks” on page 295.

- *An Introduction to Tivoli’s TME10*, SG24-4948
- *Getting Started with TME 10 User Administration*, SG24-2015
- *Tivoli User Administration Design Guide*, SG24-5108
- *A First Look at TME 10 Distributed Monitoring 3.5*, SG24-2112
- *TEC Implementation Examples*, SG24-5216
- *TME 10 Internals and Problem Determination*, SG24-2034
- *Administering IBM DCE and DFS Version 2.1 for AIX (and OS/2 Clients)*, SG24-4714
- *Protect and Survive Using IBM Firewall 3.1 for AIX*, SG24-2577

F.2 Redbooks on CD-ROMs

Redbooks are also available on CD-ROMs. **Order a subscription** and receive updates 2-4 times a year at significant savings.

CD-ROM Title	Subscription Number	Collection Kit Number
System/390 Redbooks Collection	SBOF-7201	SK2T-2177
Networking and Systems Management Redbooks Collection	SBOF-7370	SK2T-6022
Transaction Processing and Data Management Redbook	SBOF-7240	SK2T-8038
Lotus Redbooks Collection	SBOF-6899	SK2T-8039
Tivoli Redbooks Collection	SBOF-6898	SK2T-8044
AS/400 Redbooks Collection	SBOF-7270	SK2T-2849
RS/6000 Redbooks Collection (HTML, BkMgr)	SBOF-7230	SK2T-8040
RS/6000 Redbooks Collection (PostScript)	SBOF-7205	SK2T-8041
RS/6000 Redbooks Collection (PDF Format)	SBOF-8700	SK2T-8043
Application Development Redbooks Collection	SBOF-7290	SK2T-8037

F.3 Other Publications

These publications are also relevant as further information sources:

- *IBM Global Sign-On for Multiplatforms, Version 2.0 Installation and Server Management Guide*, GC32-0284, product documentation, also shipped with the product
- *IBM Global Sign-On for Multiplatforms, Version 2.0 User Administration Guide*, GC32-0285, product documentation, also shipped with the product
- *IBM DCE for AIX, Version 2.2: High Availability Cluster Multi-Processing Guide for DCE and DFS*, softcopy document shipped with the DCE for AIX 2.2 product
- OS/390 V2R5.0 Security Server (RACF) Security Administrator's Guide, SC28-1915

F.4 Web Links

The IBM Globals Sign-On home page is located at:

`www.software.ibm.com/enetwork/globalsignon`

Additional information about IBM DCE can be found at:

`www.software.ibm.com/enetwork/dce`

Information about Litronic Inc. and their products mentioned in this book can be found at:

`www.litronic.com`

Information about Biometric Access Corporation and their products mentioned in this book can be found at:

`www.biometricaccess.com`

IBM employees can access some additional information, such as presentations and white papers, from the internal GSO home page at:

`w3.software.ibm.com/sales/networkingsw/globalsignon`

How to Get ITSO Redbooks

This section explains how both customers and IBM employees can find out about ITSO redbooks, CD-ROMs, workshops, and residencies. A form for ordering books and CD-ROMs is also provided.

This information was current at the time of publication, but is continually subject to change. The latest information may be found at `http://www.redbooks.ibm.com/`.

How IBM Employees Can Get ITSO Redbooks

Employees may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **Redbooks Web Site on the World Wide Web**

  `http://w3.itso.ibm.com/`

- **PUBORDER** – to order hardcopies in the United States
- **Tools Disks**

  To get LIST3820s of redbooks, type one of the following commands:

  ```
  TOOLCAT REDPRINT
  TOOLS SENDTO EHONE4 TOOLS2 REDPRINT GET SG24xxxx PACKAGE
  TOOLS SENDTO CANVM2 TOOLS REDPRINT GET SG24xxxx PACKAGE (Canadian users only)
  ```

  To get BookManager BOOKs of redbooks, type the following command:

  ```
  TOOLCAT REDBOOKS
  ```

  To get lists of redbooks, type the following command:

  ```
  TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET ITSOCAT TXT
  ```

  To register for information on workshops, residencies, and redbooks, type the following command:

  ```
  TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ITSOREGI 1998
  ```

- **REDBOOKS Category on INEWS**
- **Online** – send orders to: USIB6FPL at IBMMAIL or DKIBMBSH at IBMMAIL

Redpieces

For information so current it is still in the process of being written, look at "Redpieces" on the Redbooks Web Site (`http://www.redbooks.ibm.com/redpieces.html`). Redpieces are redbooks in progress; not all redbooks become redpieces, and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

How Customers Can Get ITSO Redbooks

Customers may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **Online Orders** – send orders to:

	IBMMAIL	**Internet**
In United States	usib6fpl at ibmmail	usib6fpl@ibmmail.com
In Canada	caibmbkz at ibmmail	lmannix@vnet.ibm.com
Outside North America	dkibmbsh at ibmmail	bookshop@dk.ibm.com

- **Telephone Orders**

United States (toll free)	1-800-879-2755
Canada (toll free)	1-800-IBM-4YOU
Outside North America	(long distance charges apply)
(+45) 4810-1320 - Danish	(+45) 4810-1020 - German
(+45) 4810-1420 - Dutch	(+45) 4810-1620 - Italian
(+45) 4810-1540 - English	(+45) 4810-1270 - Norwegian
(+45) 4810-1670 - Finnish	(+45) 4810-1120 - Spanish
(+45) 4810-1220 - French	(+45) 4810-1170 - Swedish

- **Mail Orders** – send orders to:

IBM Publications	IBM Publications	IBM Direct Services
Publications Customer Support	144-4th Avenue, S.W.	Sortemosevej 21
P.O. Box 29570	Calgary, Alberta T2P 3N5	DK-3450 Allerød
Raleigh, NC 27626-0570	Canada	Denmark
USA		

- **Fax** – send orders to:

United States (toll free)	1-800-445-9269
Canada	1-800-267-4455
Outside North America	(+45) 48 14 2207 (long distance charge)

- **1-800-IBM-4FAX (United States)** or **(+1) 408 256 5422 (Outside USA)** – ask for:

  Index # 4421 Abstracts of new redbooks
  Index # 4422 IBM redbooks
  Index # 4420 Redbooks for last six months

- **On the World Wide Web**

Redbooks Web Site	http://www.redbooks.ibm.com
IBM Direct Publications Catalog	http://www.elink.ibmlink.ibm.com/pbl/pbl

Redpieces

For information so current it is still in the process of being written, look at "Redpieces" on the Redbooks Web Site (`http://www.redbooks.ibm.com/redpieces.html`). Redpieces are redbooks in progress; not all redbooks become redpieces, and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

IBM Redbook Order Form

Please send me the following:

Title	Order Number	Quantity

First name | Last name

Company

Address

City | Postal code | Country

Telephone number | Telefax number | VAT number

☐ Invoice to customer number

☐ Credit card number

Credit card expiration date | Card issued to | Signature

We accept American Express, Diners, Eurocard, Master Card, and Visa. Payment by credit card not available in all countries. Signature mandatory for credit card payment.

List of Abbreviations

ACL	Access Control List
ADSM	ADSTAR Distributed Storage Manager
API	Application Programming Interface
CDS	Cell Directory Service (DCE)
CICS	Customer Information Control System
CIM	Configuration Information Manager
CLI	Command Line Interface
CT-LIB	Client-Library (Sybase)
DB	Database
DCE	Distributed Computing Environment
DES	Data Encryption Standard
DLL	Dynamic Link Library
DNS	Domain Name System
DSB	Directory Service Broker (DASCOM)
DTS	Distributed Time Service (DCE)
EHLLAPI	Enhanced High Level Language APIS
ERA	Extended Registry Attributes
FTP	File Transfer Protocol
GDA	Global Directory Agent
GSO	Global Sign-On
GSSAPI	Generic Security Service API
GUI	Graphical User Interface
NACMP	High Availability Cluster Multi-Processing
HTTP	Hypertext Transport Protocol
IBM	International Business Machines Corporation
ITSO	International Technical Support Organization
LAN	Local Area Network
LC	Logon Coordinator
LDAP	Lightweight Directory Access Protocol
LSF	Logon Script File
MVS	Multiple Virtual Storage
NAC	Network Application Consortium
NIS	Network Information System (Yellow Pages)
OCI	Oracle Call Interface
ODBC	Open Database Connectivity
PCOMM	(IBM eNetwork) Personal Communications (for Windows NT and 95)
PIN	Personal Identification Number
PKM	Personal Key Manager
PTF	Program Template File
RACF	Resource Access Control Facility
RDBMS	Relational Database Management System
RPC	Remote Procedure Call
SLIP	Serial Line Internet Protocol

SMIT	Systems Management Interface Tool (AIX)
TCL	Tool Command Language
TCP/IP	Transmission Control Protocol/Internet Protocol
TEC	Tivoli Enterprise Console
TME	Tivoli Management Environment
TMF	Tivoli Management Framework
TMR	Tivoli Management Region (Tivoli)
TSO	Time Sharing Option
URL	Uniform Resource Locator
WAN	Wide Area Network
XSSO	X/Open Single Sign-On

Index

D

E

F

G

H

I

K

L

M

N

O

P

R

S

T

U

V

W

X

ITSO Redbook Evaluation

Sign On with IBM's Global Sign-On!
SG24-5122-00

Your feedback is very important to help us maintain the quality of ITSO redbooks. **Please complete this questionnaire and return it using one of the following methods:**

- Use the online evaluation form found at http://www.redbooks.ibm.com
- Fax this form to: USA International Access Code + 1 914 432 8264
- Send your comments in an Internet note to redbook@us.ibm.com

Which of the following best describes you?
_ **Customer** _ **Business Partner** _ **Solution Developer** _ **IBM employee**
_ **None of the above**

Please rate your overall satisfaction with this book using the scale:
(1 = very good, 2 = good, 3 = average, 4 = poor, 5 = very poor)

Overall Satisfaction _____

Please answer the following questions:

Was this redbook published in time for your needs? Yes___ No___

If no, please explain:

_____

_____

_____

_____

What other redbooks would you like to see published?

_____

_____

_____

Comments/Suggestions: (THANK YOU FOR YOUR FEEDBACK!)

_____

_____

_____

_____

SG24-5122-00
Printed in the U.S.A.