software AG

Natural Business Services

Natural Business Services Administration

Version 5.3.1

February 2010

Natural Business Services

Table of Contents

1   **Natural Business Services Administration**

Welcome to *Natural Business Services Administration* — your tool for setting up and managing environments for Natural Business Services applications.

This documentation does not provide information about the following topics. We assume that you are either familiar with these topics or have access to other sources of information about them:

- Natural programming language and environment
- Natural Construct
- EntireX
- Entire Net-Work

Natural Business Services Administration is intended for administrators who use the subsystem to manage network communications and business services and developers who create and maintain business services. This preface will help you get the most benefit from this documentation, as well as find other sources of information about managing business services.

This documentation explains how to invoke and use the Business Service Administration subsystem for system and application administration, as well as application development. It assumes that, as an administrator, you have extensive knowledge of Natural Business Services and its supporting products.

Natural Business Services Administration covers the following topics:

	Role of the Business Service Administration Subsystem	Describes the role of the Business Service Administration subsystem in supporting Natural Business Services. It also contains information on accessing and navigating the subsystem.
	Using the Business Service Administration Subsystem	Describes where things are in the subsystem and how to access them. Also describes the Natural Business Services Add-in.
	Defining and Managing Servers	Describes how to create and manage the servers that automate communication between business service components and EntireX.
	Using the Business Service Control Record	Describes how to use the business service control record to initiate, ping, and shut down servers, as well as how to enable Natural Security, specify a security exit, and adjust Entire Net-Work timeouts.
	Defining Steplibs and Domains	Describes how to setup the chain of step libraries and the domains. Steplib chains identify where your application libraries reside on the server; domains consist of a collection of application libraries, business services, and their associated modules.
	Defining Users and Security Groups	Describes how to define users, groups, and their associations. Users and groups are essential for creating a secure environment, whether you are using Natural Business Services security or Natural Security.
	Setting Business Service Security Options	Describes how to define security settings for domains, services, and methods on the client and in character mode (SYSBIZ). Also contains an overview of the server security options.
	Using Natural Business Services Tools	Describes the tools for managing the environment of Natural Business Services.
	Domains Supplied with Natural Business Services	Describes the business services contained in the ADMIN and INTERNAL domains, as well as those in the DEMO domain. Also explains security recommendations.
	Auditing Business Services	Describes the log and query options for business services.
	Deploying the Administration Subsystem	Describes the steps involved in deploying the Business Service Administration subsystem to run business services in production environments.
	Customizing Natural Business Services	Describes how to customize the default behavior of Natural Business Services.
	Using Custom Models with Natural Business Services	Describes how to use customized Natural Construct models to create business services.
	Appendix A: Parsing Server Startup Parameters	Describes how to use the BSSPARMN routine to parse the keywords specified in the Server Start parameter fields on the Maintain Servers panel.
	Appendix B: Additional Error Handling	Describes how to use the SPSERRN2 and SPSERRN user exits to enhance core error-processing in Natural Business Services applications.

🔵	**Appendix C: Utilities**	Describes the utility subprograms supplied with the Business Service Administration subsystem.

2   Role of the Business Service Administration Subsystem

The Business Service Administration subsystem consists of a table-driven server environment based in Natural. Typically, the Business Service Administration subsystem is used for three types of activities:

Activity	Tasks
System administration	■ Install and set up the Business Service Administration subsystem. For information, see the Natural Business Services Installation on Mainframes. ■ Deploy the Business Service Administration subsystem in one or more production or development environments. For information, see **Deploying the Administration Subsystem**. ■ Set up and maintain servers. Natural Business Services interacts with EntireX and your network to automate communication between the business service components on the client and on the server, as well as handle security checking for a business service (Natural Business Services security or Natural Security). For information, see **Defining and Managing Servers**. **Note:** These servers are not required for business service development environments. ■ Access the business service control record to adjust Entire Net-Work timeout values, enable or disable Natural Security, and specify a security exit, if required. For information, see **Using the Business Service Control Record**. ■ Monitor and adjust the performance of the Business Service Administration subsystem, such as synchronizing the security cache and cleanup queue and resetting the security cache. For information, see **Using Natural Business Services Tools**.
Security	■ Add users and groups to either Natural Business Services security or Natural Security (depending on the business service control record). This task can only be done on the server. ■ Define group access privileges to domains, services, and methods. **Tip:** Although this task can be done on the server, it is easier to do in Natural for Windows. ■ For information on defining these privileges on the client, see Define Security for Domains. ■ For information on defining these privileges on the server, see **Setting Business Service Security Options**. ■ For information on defining users and groups, see **Defining Users and Security Groups**.
Business service administration	■ Access the Business Service Administration subsystem to maintain business service definitions.

Activity	Tasks
	■ Access the Business Service Administration subsystem to define a steplib chain and domain.
	■ Access the Business Service Administration subsystem to maintain business service definitions.

> **Note:** Some administrative activities can be performed on the client. For information, see Natural Business Services Eclipse Plug-in or Natural Business Services Natural Plug-in.

This section covers the following topics:

System Administration Activities

The Business Service Administration subsystem provides dynamic, online management of tables used by servers. All data required by the servers to start, stop, and perform requested functions is maintained here. The subsystem also provides mechanisms to enhance the functionality of EntireX, such as a security mechanism, service start-up and shutdown mechanisms, and messaging mechanisms. It is also used to expose some of the EntireX parameters through table-driven maintenance functions, including defining Broker service settings.

System administration involves the following tasks:

■ Defining servers

Servers, which include attach, dispatch, and security servers, interact with EntireX and the network to automate the communication between business service components on the client and on the server. The server definitions enable the servers to make themselves known to EntireX and to configure themselves for the environments in which they are expected to run.

■ Managing business service servers

Using the Business Service Administration subsystem, administrators can start, stop, and check (ping) business service servers.

■ Viewing information logs

Logs maintained by the Business Service Administration subsystem contain updates to all system tables, information generated by running the business service servers, errors related to communication with the business service dispatch server, and security violations.

Business Service Administration Activities

Another way of looking at the architecture of a business service is to view the Business Service Administration subsystem components that participate in directing requests from the client to the appropriate method on the server. The following diagram illustrates the relationships between these components:
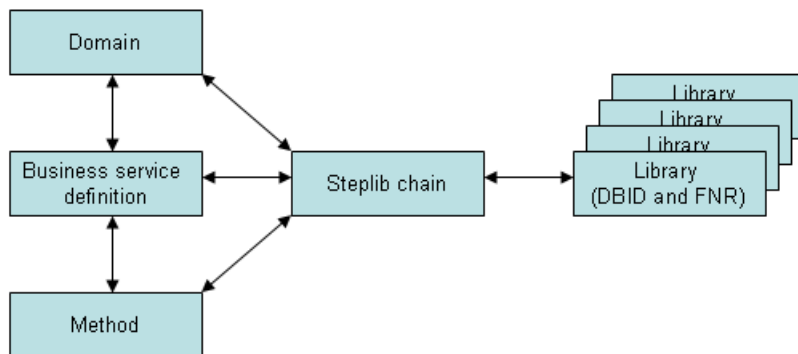
These components are:

Component	Description
Steplib chain containing step libraries	Defines a set of Natural libraries that a business service dispatch server must access to call a subprogram proxy. The dispatch server accesses the libraries in the order they are listed in the steplib definition. **Note:** For information on subprogram proxies, see Natural Business Services Subprogram-Proxy-Client Model.
Domain	Identifier used to group business services or define the boundaries of a business service. For example, all services in an Order Entry business service (such as Customer, Order, and Product) can share a single domain.
Business service definition and method	Groups a domain, business service, and version information under one identifier that describes the business service. This definition specifies all the methods of a business service, which subprogram proxies enable access to specific methods, and where the subprograms are located (steplibs).

Application administrators and developers use the Business Service Administration subsystem to define steplibs and domains. Business service definitions are also stored in the subsystem, where they can be modified as needed.

Security Administration Activities

The Business Service Administration subsystem is also used to define security settings that control user access to domains, business services, and methods.

First define the user IDs, then assign each user ID to one or more groups. Groups provide a mechanism for identifying users who need to access the same combination of domains, business services, and methods. Groups usually associate people who perform similar tasks. For example, an organization might have groups for Quality Assurance, Sales, and Management.

> **Note:** Users and groups can also be defined in Natural Security, rather than Natural Business Services security.

Once users and groups are defined in the Business Service Administration subsystem, security settings can be defined at multiple levels, as shown in the following table:

Level	Description
Domain	When a group is granted access to a domain, its members can use all services and methods in the domain.
Service	Within domains, individual services can have security settings defined so users can only access selected services.
Method	Access privileges can also be explicitly granted or revoked for individual methods of a service.

Security Cache for Natural Business Services

Because security data is located in the Business Service Administration subsystem, it can be maintained centrally. In addition, the subsystem includes a dynamically updated cache mechanism that improves throughput. When a business service dispatch server receives a request from the client, the Natural Business Services security server assigned to it checks the user's security privileges. The security server builds a security cache when it first checks the security for a given request. Thereafter, only one table lookup is required to check the access privileges of a user to requested data, thereby improving response time after the initial request.

The contents of the security cache are always dynamic. If changes occur in the Business Service Administration subsystem data that affect security, the security cache is automatically updated. The security cache is maintained as a separate encrypted table in an external data source.

3 Using the Business Service Administration Subsystem

This section explains how to invoke the Business Service Administration subsystem and illustrates the menu structure. For information on navigating menus, operating panels, and using online help, direct commands, and PF-keys, see What is Natural Construct?

Invoke the Repository Explorer in a Business Service Plug-in

You can use the repository explorer in a Natural Business Services plug-in to access the Business Service Administration subsystem. For information on invoking the repository explorer, see the documentation for the plug-in you are using.

> **Note:** For information on using the repository explorer in the Natural Business Services plug-in for Eclipse, see Use the Service Repository Explorer. For information on using the repository explorer in the Natural Business Services plug-in for Visual Studio, see Use the Business Service Explorer.

Invoke the Business Service Administration Subsystem

The Business Service Administration subsystem resides in the SYSBIZ library.

▶ **To invoke the Business Service Administration subsystem:**

1   Log onto the SYSBIZ library at the More prompt.

2   Enter "Menu" at the Command prompt.

The Business Service Administration Subsystem main menu is displayed. For example:

```
 BS__MAIN    ***** Business Service Administration Subsystem *****    CDLAYMN1
 Feb 14,06                        - Main Menu -                       08:47 PM



                    Functions


                    -------------------------------------------------


                    SA   System Administration

                    AA   Application Administration
```

```
                    ?    Help

                    .    Terminate

                    ------------------------------------------------

Function ..........  __




Command ...........  _____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      help  retrn quit        flip                                       main
```

The Business Service Administration subsystem is divided into System Administration and Application Administration functions. Each function has its own main menu, which leads to maintenance, query, and special functions. The following diagrams illustrate these menu structures.

> **Note:** If Natural Security is enabled, different options are displayed on the System Administration menus. For information, see **Use Natural Security with Natural Business Services**.

Structure of System Administration Main Menu

The following diagram shows the menu structure of the System Administration main menu:
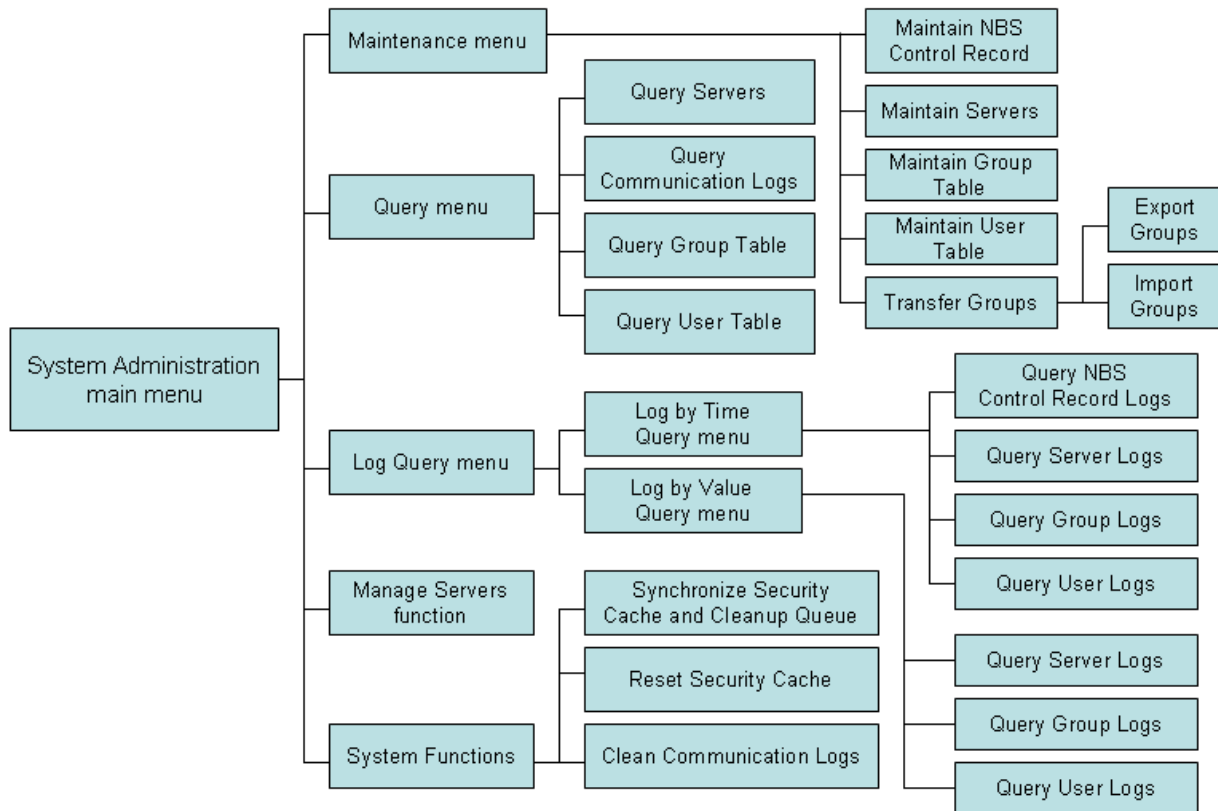
Structure of Application Administration Main Menu

The following diagram shows the menu structure of the Application Administration main menu:
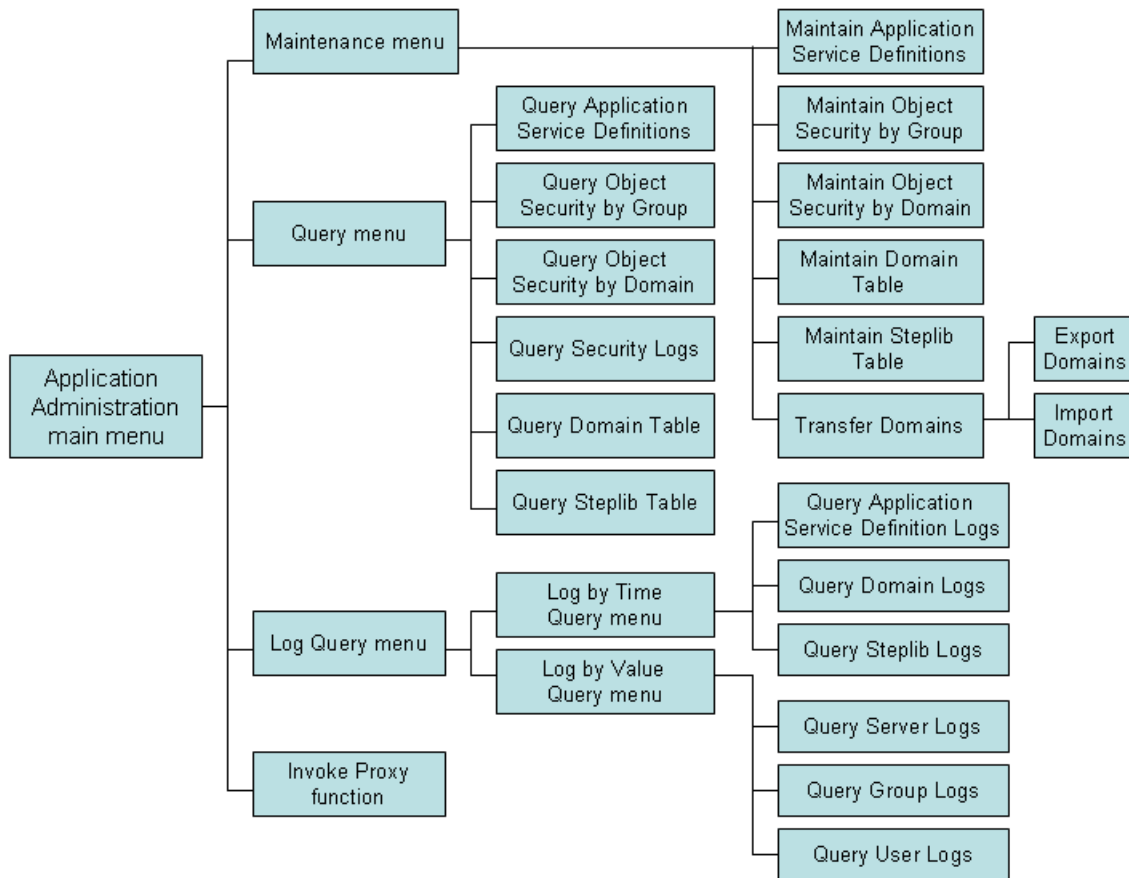
Use Natural Security with Natural Business Services

The Business Service Administration subsystem is fully integrated with Natural Security. If Natural Security is being used, all updates to group and user information are applied directly to Natural Security data. It is also possible to define user-library links. If Natural Security is not being used, the Business Service Administration subsystem manages users and their groups with its own set of security tables. The security data is always synchronized with any updates made to table data, whether the updates are made to data in the Business Service Administration subsystem or the Natural Security server.

Using Natural Security instead of Natural Business Services built-in security allows you to use security definitions that are already in place. There is no need to duplicate this information in the Business Service Administration subsystem files. This functionality is provided as an alternative for users without access to a Natural Security server.

Notes:

1. For information on enabling Natural Security in the Business Service Administration subsystem, see **Using the Business Service Control Record**.

2. For information on defining users and groups when Natural Security is enabled, see **Defining Users and Security Groups**.

This section covers the following topics:

- Configure Natural Security for Natural Business Services
- Start Services in Batch Mode under Natural Security
- Natural Security Interface to Restricted Libraries
- System Administration Menus with Natural Security

Configure Natural Security for Natural Business Services

You can use Natural Security to maintain users and groups by configuring your Natural Security environment so that it informs Natural Business Services whenever changes are made to user or group tables. This ensures the integrity of data in Natural Business Services security cache. If a user is unlinked from a Natural Security group, all records associated with that user in the security cache are also deleted.

▶ **To configure the Natural Security environment:**

1   Copy the NSCUSEX1 module from the SYSBIZ library into the SYSSEC library.

    If you are already using the NSCUSEX1 exit in Natural Security, copy the sections of code identified in the NSCUSEX1 module supplied with Natural Business Services into your existing version of NSCUSEX1.

2   Use Natural Security to add SYSBIZ as a steplib to the SYSSEC library.

3   Catalog NSCUSEX1 in the SYSSEC library.

4   Modify your NATPARM to start Natural Security sessions to include the LFILE definitions required by Natural Business Services (LFILE 135 and 136).

Start Services in Batch Mode under Natural Security

When you are starting Natural Business Services servers as a batch job under Natural Security, refer to the guidelines supplied in the Natural Security documentation. For example, use `STACK=(LOGON[library] [user-id] [pswd])` in the NATPARM settings used to start the Natural session in the batch job.

The user ID used to perform the Natural Security logon for batch jobs starting Natural Business Services must be linked to the SYSBIZ library. No other special privileges need be granted to the user.

▶ **To allow Natural Business Services to communicate with Natural Security:**

1   Invoke the Natural Security main menu.

2   Invoke the Administration Services main menu.

3   Invoke the General Options menu.

4   Set the Free access to functions via interface subprograms property to "True".

Natural Security Interface to Restricted Libraries

Natural Business Services uses the published Natural Security interface, NSC---L, to determine whether a user may use a library. The interface does not provide a means to specify a DBID and FNR with the library name. If a user is linked to a library through Natural Security on a given DBID or FNR, Natural Business Services cannot verify that the library it is accessing on behalf of the user is on the same DBID or FNR specified under Natural Security.

When you define steplib chains in the Business Service Administration subsystem, be sure to only specify libraries that are protected by Natural Security. When you specify a steplib with a DBID or FNR in Natural Business Services, define the library in Natural Security with the same DBID or FNR.

> **Note:** You cannot use the same name for libraries with different DBIDs or FNRs.

System Administration Menus with Natural Security

When Natural Security is enabled, the options on the System Administration menus differ slightly. The following diagram shows the menu structure:
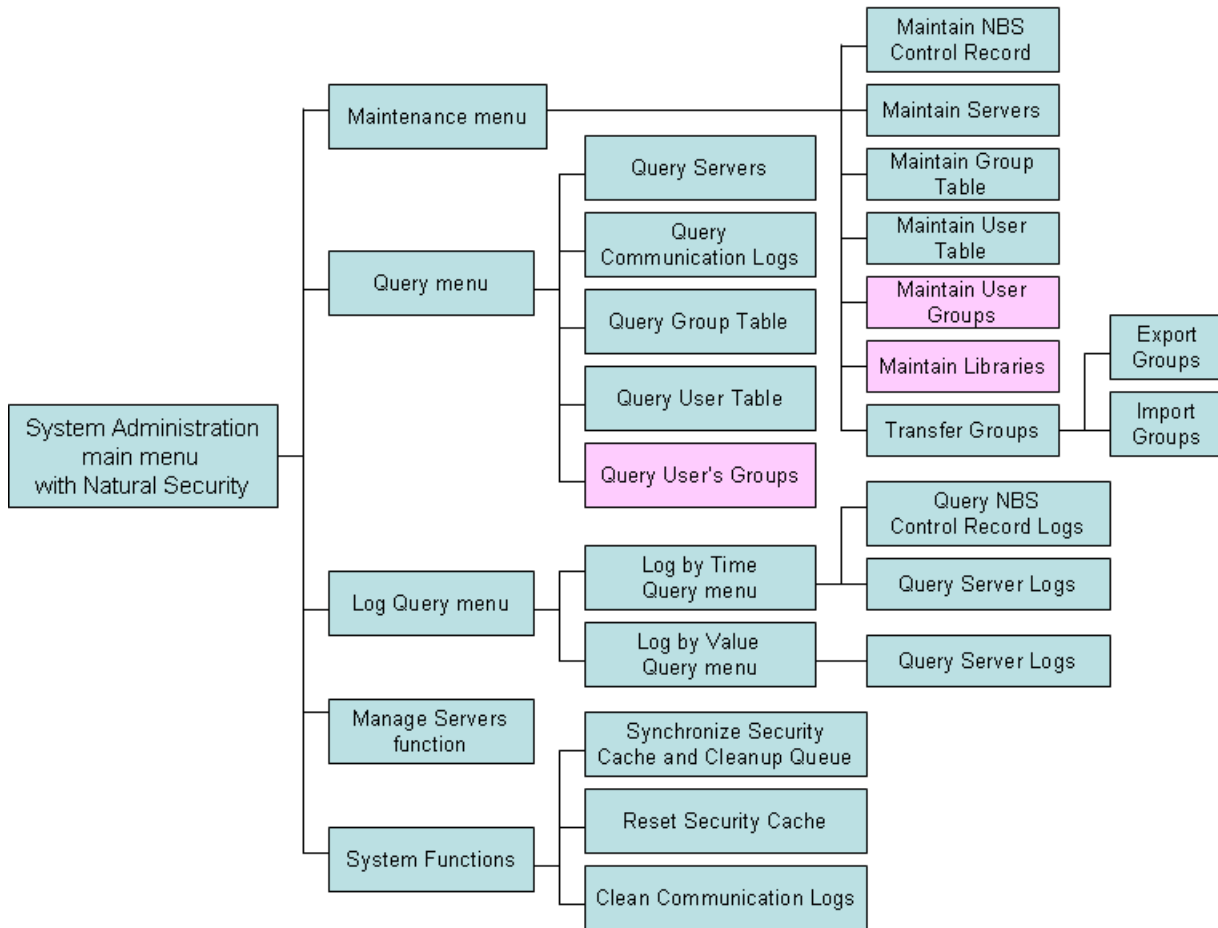
If you use Natural Security as your user and group management mechanism, the following changes are displayed in the System Administration menus:

- System Administration Maintenance menu changes include:
  - Maintain User Table function invokes the Natural Security User Maintenance panel
  - Maintain Group Table function invokes the Natural Security Group Maintenance panel
  - Maintain User Groups function is available (see diagram)

    This function allows you to associate users and groups.
  - Maintain Libraries function is available (see diagram)

    This function allows you to define links between users and libraries.
- System Administration Query menu changes include:
  - Query User Table function invokes a Natural Security User Information window
  - Query Group Table function invokes a Natural Security Group Information window
  - Query User's Groups function is available (see diagram)

■ Queries for user and group log information (by time and value) are not available

Use EntireX Security with Natural Business Services

If you use EntireX security to manage a security server (such as RACF, CA-TOP SECRET, or ACF2), you can use your established security system to authenticate users accessing business services. Natural Business Services supplies the SPSSAF module in the SYSBIZ library, which calls a Natural routine to perform authorization checks based on service, domain, method, and user ID.

▶ **To configure EntireX security for Natural Business Services:**

1   Add a new resource class or type called "NBS".

> **Note:** If you call it something other than NBS, you must change the reference to the class or type in SPSSAF.

2   Create resource profiles for each domain/service/method combination used in the application using the format: `domain.service.method`.

Ensure that the number of characters specified in resource profiles does not exceed 32. Do not use special characters in your resources.

3   Add the resource profiles to the resource class or type.

4   Set up your business service dispatch server to use the SAF security mode.

> **Notes:**

1. For information on setting up the dispatch server, see **Defining and Managing Servers**.

2. For information on defining resources to your security server, refer to the EntireX Security documentation.

💡 **Tip:** If you modify the resource profile after the business service is in use, reset the business service security cache (to clear previous permissions from memory). For information, see **Reset the Security Cache**.

Use a Hybrid Security Server with Natural Business Services

Natural Business Services also supports a hybrid security server. For example, you can use EntireX security for user authentication (mainframe access rights) and perform validations against FSEC (Natural library and domain/service/method authorization) or Natural Business Services security. The business service control record definition determines which security server is used.

▶ **To use a hybrid security server with a dispatch server:**

1   Display the second Maintain Servers panel for the dispatch server.

     For information, see **Access the Maintain Servers Panels**.

2   Enter one of the following settings for SECURITY-MODE in the Server Start Parameters section:

Setting	Description
SAF-NSC (for validations against FSEC)	The dispatch server checks business service rights defined in the Natural Security file.
SAF-APPL (for Natural Business Services security)	The dispatch server checks business service rights defined in the Natural Business Services security file.

In both cases, the dispatch server uses SAF/EXX security for user ID and password authentication.

📄   **Notes:**

1. To use a hybrid security server, the Security mode setting for the dispatch server on the client must also be set to SAF.

2. For information on the business service control record, see **Access the Maintain Control Record Window**.

4   Defining and Managing Servers

Servers automate the interaction between business service components and EntireX, freeing developers from the task of coding calls. Servers also control such application-independent facilities as security and error recovery. EntireX controls the compression, encryption, and data translation facilities.

Natural Business Services provides predefined dispatch, security, and attach servers you can modify to suit your needs. You choose how the servers are initiated (for example, in individual batch jobs or as subtasks of an attach server). You can also create your own servers.

Once you have modified or created servers for your system, you can test, initiate, shut down, and ping them (determine if they are running). You can also view query information about the servers.

> **Note:** If you are using SPoD for development only, you do not need to define and manage the servers until the code is required at runtime (for example, to execute a business service from Visual Studio, Java, or a Web service).

This section covers the following topics:

Types of Servers

The three types of servers are:

Server Type	Description
Attach	Server that starts other servers. When defining dispatch, security, or user-defined servers, you can associate them with an attach server. If a server is intended to execute as a subtask, the attach server must be executing before the servers associated with it can be started. Attach servers can be used to automatically start other servers on demand. **Note:** For more information, see **Attach Servers**.
Server	Program that registers a service with EntireX and enters a receive loop waiting for clients to request its services. Initiating a server causes the common start facility (SPSSTART) to automatically logon and register with EntireX. Servers can also act as Broker clients. Servers implement two kinds of interfaces: main and command. **Note:** For more information, see **Main and Command Interface Services**.
Client	User of services. Client servers do not register services with EntireX, but request the use of services by sending a message to a server. When clients are started, no implicit commands are executed. Normally, client servers are only used to define a Broker ID, user ID, and password required to logon to EntireX.

Main and Command Interface Services

All servers implement two interface services: main and command. The main interface service is defined on the Maintain Servers panel, where the service field always contains the value, MAIN. The command interface service uses the same server class and server name, but it uses the CMD Broker service name. For example:

Interface	Server Class	Server Name	Broker Service Name
Main	BUSINESS	DISPATCH	MAIN
Command	BUSINESS	DISPATCH	CMD

> **Note:** The Server class, Server name, and Broker service name must be defined in the Broker attribute file.

The main interface service implements commands that are specific to the server. The command interface service implements commands that are independent of servers; that is, commands that are generic and applicable to all servers, such as PING and SHUTDOWN. Having separate servers allows you to automate the server startup process in response to some server services but not others. For example, a dispatch server should be automatically started when a dispatch request is received, but commands such as PING and SHUTDOWN should not cause a new server to be launched.

The command interface services are:

Command	Description
PING	Determines whether a server is available.
ENVIRONMENT	Retrieves information about the environment in which the server is running.
TRACE	Sets trace options to help debug a server or service.
CALLNAT	Executes a user-supplied subprogram.
SHUTDOWN	Ends a server.
ERROR	Triggers a runtime error in the server.
STEPLIBS	Lists the steplibs associated with the server.
SYSPROF	Returns information about the system files used by a server.

> **Note:** When sending command interface services programmatically, use the syntax: `CMD` *command*, where *command* is one of the commands listed above.

Define Servers

Use the Maintain Servers panels in the Business Service Administration subsystem to define servers. You can create the following types of servers:

- attach

- server

- client

This section describes how to access and use the Maintain Servers panels. It also describes how to add information to the Broker attribute file and create a Natural profile to initialize a server at runtime.

The following topics are covered:

- Access the Maintain Servers Panels
- Identify the Server
- Define Broker Service Settings
- Define Server Start Parameters and Routines
- Define Server Settings and Program Parameters
- Change the Broker Attribute File
- Add a Natural Profile

Access the Maintain Servers Panels

▶ **To access the Maintain Servers panels:**

1   Enter "SA" in `Function` on the Business Service Administration Subsystem main menu.

     The System Administration main menu is displayed.

2   Enter "MM" in `Function` on the System Administration main menu.

     The System Administration Maintenance menu is displayed.

3   Enter "SE" in `Function` on the System Administration Maintenance menu.

The first Maintain Servers panel is displayed. For example:

```
SPBSMD     ***** Business Service Administration Subsystem *****      SPBSMD11
Feb 14,06                    - Maintain Servers -                     1 more >


 Action (A,B,C,D,M,N,P)    _ Name: _____

                            Desc: _____

+- Broker Server Settings ------------------------------------------------+
| Broker ID...............: _____                   |
| Server class............: _____                   |
| Server name.............: _____                   |
| Broker service name.....:                                                |
| Disable trusted user....: _    **Watch size and case for userid password |
| User ID.................: _____                    |
| Password................:                                                 |
| Attach Server name......: _____ * Attach..: _  |
+- Server Start Parameters -----------------------------------------------+
| >                                                                        |
| >                                                                        |
| >                                                                        |
| Server start routine....: _____ *      Source        Object           |
+--------------------------------------------------------------------------+
Command: _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
confm help  retrn quit       flip  pref                     left  right main
```

4   Provide a name for the server and specify the Broker server settings.

For information, see **Identify the Server** and **Define Broker Service Settings**.

Optionally, you can specify the following settings; not all of these settings are appropriate to all types of services:

- Disable a trusted user

- Provide a user ID and password for the broker

- Designate an attach server and provide the attach server name. Servers that are linked to an attach server must use the same Broker ID as the attach server. When the Broker ID for an attach server is modified, the Broker ID for all related servers will be updated automatically.

> **Note:** Broker service settings configure the server to register itself by a specific name with a specific broker. You must define settings for servers of type A (attach) and S (server).

5   Assign a server start routine for the server.

For information, see **Use Server Start Routines**.

6   Press PF11 (right).

The second Maintain Servers panel is displayed. For example:

```
SPBSMD     ***** Business Service Administration Subsystem *****      SPBSMD21
< 1 more                      - Maintain Servers -                      8:52 PM

 Action (A,B,C,D,M,N,P)    _ Name: _____
                             Desc: _____
+- Server Settings ----------------------------------------------------------+
| Server type..............: _ (Server, Attach server, Client)               |
|                                                                            |
| Default language........: __     Server Version.....: ___                  |
|                                                                            |
| Server timeout..........: _____                                            |
|                                                                            |
| Unique ID...............:                                                  |
+----------------------------------------------------------------------------+
+- Program Parameters -------------------------------------------------------+
|>                                                                           |
|>                                                                           |
|>                                                                           |
| Natural program name....: _____          Source        Object          |
+----------------------------------------------------------------------------+
Command: _____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
confm help  retrn quit        flip  pref                    left  right main
```

7   Define the server timeout value, default language, and program parameters.

These settings configure the server internally. Depending on the server type, you must define some or all of these settings. For information, see **Define Server Settings and Program Parameters**.

8   Press Enter to confirm the settings.

Identify the Server

Use the following fields on the first Maintain Servers panel to identify and describe the server:

Field	Description
Name	Unique identifier for the server.
Desc	Brief description that assists you in identifying the server. Be sure to make the description distinctive, since it is used on other panels to help identify the server.

Define Broker Service Settings

Use the following fields on the first Maintain Servers panel to specify the parameters the server needs to register with EntireX:

Field	Description
Broker ID	Broker identifier. Each installation of EntireX is assigned a Broker ID. This number uniquely identifies EntireX to your network. If you do not know the Broker ID, see your organization's network administrator.
	Specify the ID number for the broker being used. For example, if you are using EntireX designated as number 45, the Broker ID must contain 045 somewhere in the ID, such as BKR045, BID045, or B045NODE. Ensure that the Broker ID only contains one such numeric component.
	Note: If a type S server specifies an attach server, there is no need to supply a Broker ID because it is automatically assigned to match the Broker ID for the attach server.
Server class	Name of the Broker class used to logically organize the environment in which the servers are running. For example, BUSINESS-PROD for production services or BUSINESS-TEST for services running in a test environment.
Server name	Name of the Broker server used to logically describe a server (rather than the name of the program that implements the server). This allows you to change the program name without affecting the client programs that use the service. For information, see **Supplied Server Names**.
Broker service name	Name of the Broker service. This field is always assigned the value, MAIN. For information, see **Main and Command Interface Services**.
	The Server class, Server name, and MAIN and CMD Broker service names must be defined in the Broker attribute file for the specified broker. For information, see **Change the Broker Attribute File**.
User ID and Password	Values corresponding to values the server will use to assign the corresponding fields in the EntireX control block when making calls using the EntireX ACI (Advanced Communication Interface).

Field	Description
Attach server name	Name of the attach server used for type S servers. ■ If the server is initiated automatically on demand, specify the name of the attach server used to launch the server and mark `Attach`. ■ If the server is initiated manually, only specify the name of the attach server; do not mark `Attach`. **Note:** For information on attach servers, see **Attach Servers**.
Attach	If this field is marked, the server is initiated automatically on demand. If this field is blank, the server is initiated manually.

On startup, a server uses these parameters to tell EntireX that it is running and ready to handle requests. Later, you will add some of this information to the Broker attribute file.

Note: For information, see **Change the Broker Attribute File**.

Supplied Server Names

The following table shows the server names supplied with Natural Business Services, along with the name of the Natural program that implements the service:

Server Name	Natural Program	Description
ATTACH	SPSATT	Attach server
DISPATCH	SPSDISP	Dispatch CALLNAT requests
SECURITY	SPSSEC	Security server
TIMESTAMP	SPSTIMS	Timestamp demo service

Attach Servers

EntireX supports an automated server startup mechanism by means of an attach server. An attach server typically registers its own service with EntireX and then registers other servers it is capable of starting using an attach option. When a request is made for an attach server and no server is available (waiting in a receive state), EntireX notifies the attach server that a request is pending. Normally, that attach server launches a new server to satisfy the request.

This section covers the following topics:

■ Link Servers to an Attach Server
■ Activation Policies

- Runtime Error Handling

Link Servers to an Attach Server

An attach server must be defined on the second Maintain Servers panel using Server type A (attach). All servers that the attach server automatically launches must specify the name of the attach server and have `Attach` marked on the first Maintain Servers panel. This tells the attach server to register the current main interface service with EntireX using the Attach option.

> **Note:** For servers that are only to be launched manually, specify the attach server name without marking `Attach`.

The combination of Server class, Server name, and Broker service name must be unique for all servers that are linked to an attach server and have `Attach` marked.

An attach server can launch servers as separate batch jobs or as subtasks of the attach server. For information, see **Use Server Start Routines**.

> **Note:** The attach server only registers the main interface for servers it is capable of launching. Therefore, commands such as PING and SHUTDOWN (which go to the command interface) will not cause a server to be launched.

Activation Policies

Each time EntireX calls an attach server to satisfy an attach request, the attach server launches one instance of the requested server. This server continues to run indefinitely as long as it is satisfying server requests. If the server stays idle for longer than its timeout period (specified on the second Maintain Servers panel), it shuts down.

Runtime Error Handling

If a server receives a runtime error, its restart policy depends on whether it is linked to an attach server. If the server is not linked to an attach server, it attempts to recover from the error. Otherwise, the server shuts down as a result of the error.

Common Server Startup Interface

The main method of starting servers is using the START command. This section describes the different parameters you can use with the START command, depending on the type of server and the method of initiating it.

For servers of type S or A, the START command takes care of logging onto EntireX and registering the required Broker services. It then invokes the Natural program specified on the second Maintain Servers panel.

This section covers the following topics:

- Use the START Command to Initiate Servers as Subtasks
- Use the START Command to Initiate Servers Online or in Batch Mode

Use the START Command to Initiate Servers as Subtasks

When you are using an attach server to initiate a server, the START command does not require any parameters. The system automatically uses the name of the server being defined. For example, the following parameters can be used to start the current server as a subtask of an attach server:

```
Server Start Parameters
> NATPARM='FNAT=(1,2),PROFILE=SYSBIZ
> STACK=(LOGON SYSBIZ;START)'
>
 Server start routine...: SUBTASKB *        Source SYSBIZ   Object SYSBIZ
```

Use the START Command to Initiate Servers Online or in Batch Mode

To start a server online (from the Next prompt) or in an individual batch job, the START command must have at least one parameter: the name of the server to use for configuration on startup. For example:

```
START DISPATCHER
```

where `DISPATCHER` is the name of the server definition used for configuration.

> **Note:** The START command is a front-end to the SPSSTART program. When running servers from outside the SYSBIZ library, use SPSSTART rather than START to ensure access to the proper environment.

Define Server Start Parameters and Routines

Most servers are assigned a start routine to initiate the server in an individual batch job or through an attach server. The server start parameters and start routine fields are located at the bottom of the first Maintain Servers panel. The server start parameters differ depending on which server start routine is used: BATCHJOB or SUBTASKB.

Although the server start parameters are shown on three lines, internally these lines are treated as one long string value. To define parameters that span multiple lines, continue the parameter on the next line. For example:

```
Server Start Parameters
> JCL=BATCHTXT,JOB-PREFIX=AS,NATPARM='IM=D,PROFILE=SYSBIZ,
> STACK=(LOGON SYSBIZ,MYUSERID,MYPWD)'
>
Server start routine...: BATCHJOB *        Source SYSBIZ   Object SYSBIZ
```

We recommend that you use commas to delimit parameters, although you can also use blank characters. If a parameter ends on the last character of a line, and it is not to be concatenated with

the parameter beginning on the next line, include a comma (or blank character) at the beginning of the next line.

This section covers the following topics:

- Use Server Start Routines
- BATCHJOB Server Start Routine
- SUBTASKB Server Start Routine

Use Server Start Routines

All servers must be assigned a start routine (specified in `Server start routine` on the first Maintain Servers panel). The start routine defines how the Natural session that will host the server is initiated. Two start routines are supplied with Natural Business Services in source form. You can use one or both of these in most environments, or use them as examples for writing your own routine.

The supplied start routines are:

Start Routine	Description
BATCHJOB	Calls NATRJE to launch a batch job to host the server.
SUBTASKB	Calls CMTASK to start a Natural subtask. This routine can only be used in Natural environments that support subtasks. In such environments, BATCHJOB is typically used to start attach servers and SUBTASKB is used to start other servers.

BATCHJOB Server Start Routine

The BATCHJOB server start routine launches a separate Natural batch job to host the server. All attach servers are started using this routine.

A Natural source member in the SYSBIZ library contains JCL templates you can copy and modify to create a customized BATCHTXT member. These templates are:

- BATCHJCL (z/OS)

- BATCHDCL (z/VSE)

- BATCHBCL (BS2000/OSD)

> **Note:** For more information, see Activate the Business Service Administration Subsystem.

The server start parameters must specify the name of the source member that contains the JCL template. The source member is identified by the JCL keyword and must exist in the library where the attach server that launches the server is running (usually the SYSBIZ library).

In most cases, you can use the default values defined in the BATCHJOB program or hardcoded in the JCL. For example, you can use the following parameters to start up a batch job:

```
JCL=BATCHTXT,JOB-PREFIX=XX
```

You can insert &*keyword* place holders in the JCL templates to create the JCL member. The place holders will be replaced with the keyword values. The keywords are:

Keyword	Description
JOB-PREFIX	Two-character prefix used with JOB-NAME to associate the job with the server. If JOB-NAME is to be uniquely generated, you can specify a two-character JOB-PREFIX keyword for each server. This value defaults to SP.
JOB-NAME	Name of the job. Use the JOB-NAME keyword to explicitly name the job. In the JCL template, the &JOB-PREFIX&JOB-NR keyword is used to represent JOB-NAME. **Note:** Some operating systems do not allow multiple jobs with the same name to be run concurrently. Therefore, using the JOB-NAME keyword rather than the JOB-PREFIX keyword may cause a conflict.
TRANSACTION	Name of the Natural nucleus under which the server runs. This nucleus must be linked to Natural Business Services load modules. The default transaction is NATBAT. You can also hardcode the desired transaction within the JCL member. **Note:** You can change the transaction name using CSRLOAD. For information, see Activate the Business Service Administration Subsystem.
NATPARM	Natural parameters used when initiating the server. The default NATPARM is: `'IM=D, PROFILE=SYSBIZ'`.
LIBRARY	Name of the library to which you want to logon. This value defaults to SYSBIZ.
USER-ID	Natural user ID used to logon to Natural. This keyword only applies to servers running under Natural Security. If the USER-ID value matches the user ID specified in the server settings, use *USER-ID.
PASSWORD	Natural Security password associated with USER-ID. If this password matches the one specified in the server settings, use *PASSWORD (which prevents the password from being displayed on the current panel).
PROGPARM	If the invoked server program accepts additional parameter values, they can be substituted for the PROGPARM keyword. These parameters can also be re-substituted with parameter values passed into SPCATT, which is used to launch a remote server from a client.

The following example supplies a full JOB-NAME, rather than just the prefix. It also overrides the NATPARM. Because the statement contains special characters, it must be embedded within quotes. For example:

```
JCL=MYJCL,JOB-NAME=TESTJOB,USER-ID=*USER-ID,PASSWORD=*PASSWORD,
NATPARM='IM=D,FNAT=(1,2),PROFILE=QA'
```

Run Under Natural Security

If the server runs under Natural Security, you can include a user ID and password. The following example uses the user ID and password specified in the Broker service settings:

```
Service.................: MAIN
User ID.................: TECH1
Password................:
```

Initiate Multiple Dispatchers During Startup

During startup, the attach server can automatically initiate instances of another server (typically the dispatcher).

▶ **To initiate multiple dispatchers during startup:**

1   Add continuation characters after the `&Q-CONV-ID` parameter in the JCL member.

2   Insert two lines.

3   Type the logical name of the server to be initiated by the attach server on the first line.

    Include the continuation characters.

4   Type the number of instances of the server to initiate on the second line.

    For example, see lines 0890, 0900, and 0910 below:

```
0770 LOGON &LIBRARY
0780 SPSSTART
0790 &BUSINESS-SERVICE,%
0800 &USER-ID,%
0810 %*
0820 &PASSWORD,%
0830 &TRACE-OPTION,%
0840 &TRACE-LOCATION,%
0850 &Q-BROKER-ID,%
0860 &Q-USER-ID,%
0870 &Q-TOKEN,%
0880 &Q-SECURITY-TOKEN,%
0890 &Q-CONV-ID,%
0900 DISPATCH,%
0910 5
0920 &PROGPARM
0930 FIN
```

> **Note:** No error will occur if these two values are not specified.

SUBTASKB Server Start Routine

The SUBTASKB server start routine accesses, by means of EntireX, an attach server running in batch mode. The request to start the specific task is transmitted to the attach server. The attach server uses the data to start the requested server as an attached task inside the batch region.

Use this routine if you want the server to be executed as a subtask of an attach server. You must also specify an attach server name. To launch the current server, the attach server must be running.

> **Note:** For operating systems that support Natural's subtasking feature, this is the preferred approach because fewer batch jobs need to be initiated.

When using SUBTASKB, specify the parameters required to start the subtask. The STACK parameter should include a LOGON to the SYSBIZ library and the required stack program. To only launch the current server, use the START keyword.

> **Note:** The STACK parameter differs depending on whether servers run under Natural Security or which AUTO parameter is specified in your NATPARM.

The SUBTASKB server start routine supports the following keywords:

Keyword	Description
TRANSACTION	Name of the Natural nucleus under which the server runs. This defaults to the same transaction as used by the attach server that launches the substask.
NATPARM	Natural parameters used to initiate the server. The parameters can be any value allowed when initiating a Natural session.

This section covers the following topics:

- Sample SUBTASKB Routines
- Write to Print and Work Files
- Generate Trace Output

Sample SUBTASKB Routines

The NATPARM is usually the only required parameter. The format of this parameter varies depending on whether or not you are using Natural Security.

The following example shows the SUBTASKB parameters without Natural Security:

```
Server Start Parameters
> NATPARM='FNAT=(1020,1000),PROFILE=SYSBIZ,
> STACK=(LOGON SYSBIZ;START)'
```

Because it contains special characters, the NATPARM parameter must be within quotes. In the example above, the START keyword invokes the common server startup facility. Once Natural is started, the Natural program name defined on the second Maintain Servers panel is invoked.

Under Natural Security, the Natural startup parameters are based on the value of the AUTO parameter in the NATPARM. The following example uses AUTO=OFF so that the user ID and password must be supplied. You can do one of the following:

- Enter a valid Natural user ID and password
- Enter the *USER-ID and *PASSWORD keywords to use the values specified in the Broker Service Settings portion of the first Maintain Servers panel. For example:

```
Server Start Parameters
> NATPARM='FNAT=(1,2),PROFILE=SYSBIZ,AUTO=OFF,
> STACK=(LOGON SYSBIZ,*USER-ID,*PASSWORD;START)'
```

Write to Print and Work Files

Programs executed by a dispatch server running in batch mode can write information to a print or work file. When doing so, the batch job that initiates the server (the attach job if you are using an attach server) must define the necessary output devices. For example: CMPRT*nn* and CMWKF*mm*.

Generate Trace Output

If you experience problems initiating or running servers, you can trace the flow of the Natural Business Services software to help uncover the problem or to provide additional information to Software AG support. To generate trace output, stack the TRACE command prior to initiating the desired server.

> **Note:** If you intend to use the Trace function, printer 2 and 3 must be assigned to batch mode. For information, see Define the NATPARM.

The following example shows the parameters to enable tracing:

```
Server Start Parameters
> NATPARM='FNAT=(1,2),PROFILE=SYSBIZ,
> STACK=(LOGON SYSBIZ;TRACE;START)'
```

You can also pass a number from one to five to the TRACE command, which causes the trace output to be written to CMPRT01 through CMPRT05, respectively. Or, you can pass W1 through W5, which causes the trace output to be written to CMWKF01 through CMWKF05, respectively.

> **Note:** You can also write additional server start routines that are applicable to other environments. Use the BATCHJOB and SUBTASKB programs as templates.

Define Server Settings and Program Parameters

Use the second Maintain Servers panel to define the server settings used internally to configure the server (for an example of this panel, see **Access the Maintain Servers Panels**). The fields on this panel are:

Field	Description
Server type	Type of server. Specify one of the following: ■ A (attach) ■ S (normal) ■ C (client) The START program automatically performs a broker Logon and Register for type A or S servers.
Default language	Value assigned to the Natural *Language system variable on server startup. The server uses this value to determine which language to use when returning messages to the client. The client can override this value when making a request, but the server always returns to the default value after servicing a client.
Server timeout	Amount of time the server will continue to run after the last time it serviced a client request. Normally, attach servers should never time out, as they are required to launch all other servers. Therefore, specify a high timeout value, such as 9999H, for attach servers. The problem with allowing other servers to time out is that users may experience a slight delay whenever a new server must be started to satisfy a request. But high timeout values may result in many idle servers consuming system resources. For most servers, 30 to 60 minutes is recommended. The server timeout value is specified as a numeric value or as a number suffixed with an interval. If there is no interval suffix, the number is assumed to specify seconds. Valid interval suffixes are: ■ S (seconds, for example, 3600S) ■ M (minutes, for example, 60M) ■ H (hours, for example, 3H)
Unique ID	16-character unique ID. This derived value is used as the prefix for the TOKEN field in the Broker control block. This allows Natural Business Services to associate a running server with a server record.
Program Parameters	Additional parameters to allow servers to configure themselves. The following program parameters are required by the supplied servers:

	Dispatch server	`SECURITY-SERVER=`*`server-name`* Name of the server that the dispatch server communicates with to retrieve security information.

Field	Description	
		`SECURITY-MODE=`*`value`*
		Parameter used to disable/enable security checking by specifying the security mode. When security is disabled, the security server is only used to look up program names and library information. When enabled, the security server uses the security mode to initiate the security check. This parameter can have the following values:
		■ NONE
		No security; program names and libraries are checked
		■ APPL
		Natural Business Services authenticates users and checks authorizations and application-level security
		■ NSC
		Natural Security authenticates users and checks authorizations
		■ SAF
		EntireX authenticates users and checks authorizations
		■ SAF-APPL or SAF-NSC
		Hybrid security server is used (for information, see **Use a Hybrid Security Server with Natural Business Services**)
		By default, SECURITY-MODE=NONE. If you change the SECURITY-MODE value, clear the security cache record.
	Attach server	`CONVERSATION-FACTORY=`*`server`*
		This parameter identifies the server record that defines the Broker settings used to generate conversations when using the SQU* API.
Natural program name	When the START command is used, this parameter identifies the program invoked by the common start facility to implement the server.	

Change the Broker Attribute File

The values specified in the Server class, Server name, and Broker service name fields on the first Maintain Servers panel must also be defined in the Broker attribute file for each server. EntireX requires these attributes to identify servers and their characteristics. In addition, you must add the Broker server definitions required by the command interface services. These are the same as the main interface services, with the Broker service name of CMD instead of MAIN.

> **Notes:**

1. Each EntireX Broker has its own attribute file, so be sure to add the Broker service definition to the appropriate attribute file.

2. You do not have to change the Broker attribute file until you start the server. Before doing that, someone with access authority must update the file. For information, refer to the EntireX documentation.

Add a Natural Profile

You can use Natural profiles to fully define a Natural environment. They are useful for reducing the amount of information you must supply when starting a Natural session. These profiles can be set up using the Natural SYSPARM utility.

Profiles can be useful when defining servers in the Administration subsystem. As part of defining a server, specify the server start parameters. Use the NATPARM keyword to identify the system file used to store the Natural profile.

> **Notes:**

1. For information on the SYSPARM utility, refer to the Natural Utilities documentation.

2. For an example of creating a Natural profile, see Natural Business Services Installation on Mainframes.

Manage Servers

Use the Manage Servers panel to initiate attach servers. Since all other servers can be initiated automatically (on demand) by an attach server, it is usually not necessary to initiate other servers in advance of their use.

When servers fail to start, you can also use the Manage Servers panel to help diagnose the problem. Additionally, you can use this panel to send inquiry commands to various servers to determine the environment in which they are running. Finally, you can use the Manage Servers panel to shut down servers.

> 📄 **Note:** You can also use a GUI application called the Server Monitor to manage servers. For information on using the Server Monitor, see NBS Server Monitor.

▶ **To access the Manage Servers panel:**

1   Enter "SA" in Function on the Business Service Administration Subsystem main menu.

The System Administration main menu is displayed.

2   Enter "MS" in Function on the System Administration main menu.

The Manage Servers panel is displayed. For example:

```
SPCMS      ***** Business Service Administration Subsystem *****      SPCMS0
Feb 14,06                     - Manage Servers -                      8:58 PM


Action Server                               Type     Attach Manager

------ -------------------------------- ------- ------------------------------
 ___     ATTACH-51-EXX                     Attach

 ___     ATTACH-51-EXX-SECURE              Attach

 ___     DISPATCH-51-EXX                   Server

 ___     DISPATCH-51-EXX-BATCH             Server  ATTACH-51-EXX

 ___     DISPATCH-51-EXX-SECURE            Server  ATTACH-51-EXX-SECURE

 ___     DISPATCH-51-EXXTEST               Server

 ___     SECURITY-51-EXX                   Server

 ___     SECURITY-51-EXX-BATCH             Server  ATTACH-51-EXX

 ___     SECURITY-51-EXX-SECURE            Server  ATTACH-51-EXX-SECURE

Server .....: _____ Server type ..: _

Replica ID .: _____ Repeat command: _

Last command:

Respondent .:
```

```
Direct Command: _____

Debug       EndJob      EndService EnVironmnt Initiate    Ping        Refresh

SHutdown    STeplibs    SYsprof     (PF4=CLog) (PF5=flip)
```
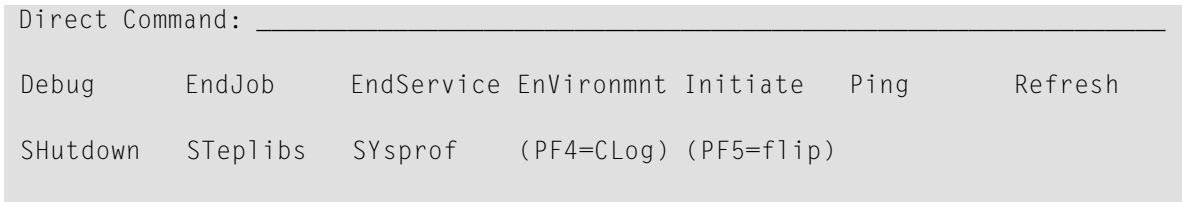
Use this panel to send commands to servers, display the replies returned from the servers, or initiate new servers. By default, this panel displays Type S (server) and A (attach) servers.

3   Use the following fields to specify options for servers:

Field	Description
Action	Indicates the command to be sent to the corresponding server. To send a command to a server, enter the upper case letter(s) of the command (shown at the bottom of the panel) in the appropriate Action field. These commands can also be used in the Broker Driver program.
Server	Repositions the displayed information to the server specified in this field.

Server type	Indicates the type of server. Do one of the following:
	
	■ Enter "*" to show all servers.
	
	■ Enter the code for the type of server you want to display. Valid codes are:
	
	  ■ A (attach)
	
	  ■ S (server)
	
	  ■ C (client)
	
	■ Leave the field blank to show servers of type A and S.

Replica ID	Identification for each unique server. EntireX allows many different servers to provide the same Broker services. Each unique server is known as a replica and is identified by a unique 32-character replica ID. For most requests, the client does not care which replica satisfies the request. Occasionally, you may want to target a specific replica (for example, to run a server online to output debugging statements such as Input, Print, etc.). When testing is completed, you will want to shut down the replica. This can be done by including the desired replica ID as an option on the SHUTDOWN command. For example, using the SHUTDOWN command for a server without specifying a replica ID shuts down any replica offering the server. Specifying a replica ID and using the SHUTDOWN command causes all other servers to ignore the command. You must continue to send the command until the desired respondent replies. All commands support a replica ID option and the Respondent field indicates which replica responded to the command. Information about a replica can be retrieved using the ENVIRONMENT command.

Field	Description
Repeat command	Number of times to repeat the command. You can execute the command multiple times. For example, to initiate three dispatch servers, type "I" beside the dispatch server and enter "3" in this field. You can determine the number of servers that are running by repeating a PING command and analyzing the replica IDs of the respondents. **Tip:** A better method of determining the number of servers available is by using the EntireX Control Center.
Last command	Text that was sent to the server to execute the most recent command. This string could also be sent to the server programmatically or using the Broker Driver program.
Respondent	Server and replica ID of the server that responded to the most recent command. The respondent is followed by the response text. **Note:** Commands are always sent to the Broker service associated with the specified server. If multiple servers specify the same Broker service, the command may be directed to an alternate server.

You can also access the communications logs to view system messages. For information, see **Access the Communications Logs**.

This section covers the following topics:

- Overview of Server Commands
- DEBUG Command
- ENDJOB Command
- ENDSERVICE Command
- ENVIRONMENT Command
- INITIATE Command
- PING Command
- REFRESH Command
- SHUTDOWN Command
- STEPLIBS Command
- SYSPROF Command

- Access the Communications Logs

Overview of Server Commands

Some commands are sent directly to the main interface (MAIN) for the associated server. Other commands are sent to the command interface (CMD) for the server. Still others are sent to the attach server associated with the server.

The following table summarizes the commands and how they are processed:

Command	Applies to:	Sent to:
DEBUG	Services of type S or C that are linked to an attach server.	MAIN interface for the attach server.
ENDJOB	Attach servers that launch subtasks.	MAIN interface for the attach server. The attach server sends SHUTDOWN requests to linked servers.
ENDSERVICE	Servers of type S that are linked to an attach server.	MAIN interface for the attach server. The attach server sends SHUTDOWN requests to the specified server.
ENVIRONMENT	All servers of type S or A.	CMD interface for the server.
INITIATE	All server types.	If the service has an attach server, the command is sent to the attach server; otherwise, the server start routines are called directly to initiate the service.
PING	All servers of type S or A.	CMD interface for the server.
REFRESH	All servers of type A.	MAIN interface for the attach server.
SHUTDOWN	All servers of type S or A.	CMD interface for the server.
STEPLIBS	All servers of type S or A.	CMD interface for the server.
SYSPROF	All servers of type S or A.	CMD interface for the server.

The Attach Server column shows the name of the attach server capable of starting the server. Servers showing highlighted attach server names are those that will automatically start in response to an EntireX attach request. The following sections describe these commands in detail.

DEBUG Command

Use this command to help diagnose problems in starting servers.

The DEBUG command only applies to servers that use an attach server. The DEBUG command returns information related to the Natural startup process. When you use the START command, the START program detects if it is in debug mode and writes out additional diagnostics. As the server is not intended to start up in debug mode, the server program is not FETCHED.

How the DEBUG Command Works

When a server is launched in debug mode, the CMPRINT dataset (where Natural writes initialization errors) is mapped to work file 7 (CMWKF07). After launching the server, the attach server pauses for three seconds and attempts to read the contents of work file 7. The contents of the file are sent back to the client.

> **Tip:** If you experience I/O errors when using the DEBUG command, try repeating the command.

Debug Unix Batch Servers

The process of debugging batch servers can be difficult, as they run in the background in non-interactive mode. Whenever they fail to start up or respond to client requests, the last resort is to analyze the output written to the CMPRINT file assigned to the Natural batch session. However, if all Natural Business Services servers run with the same CMPRINT file, they will overwrite each other's output, possibly ruining the debugging information. To prevent this problem, CSRLOAD generates server definitions with a dedicated CMPRINT file for each Natural Business Services server.

> **Tip:** To prevent multiple instances of the same server from overwriting each other's CMPRINT output, we recommend that you only initiate one request at a time while debugging.

ENDJOB Command

The ENDJOB command only applies to attach servers. More specifically, it applies to attach servers that launch other services as subtasks. When the ENDJOB command is sent to the attach server, the attach server sends SHUTDOWN commands to each server it has launched.

As part of the SHUTDOWN command, the attach server includes the PARENT-RID option so only subtasks of the attach server respect the shutdown request.

Normally, the ENDJOB command causes the batch job that hosts the attach server and its subtasks to end. However, there may still be subtasks busy processing requests and therefore unable to receive the shutdown notification. In this case, the batch job continues to run until all subtasks time out or are shut down explicitly.

ENDSERVICE Command

The ENDSERVICE command can only be used with servers that have an attach server. In response to this command, the attach server continues to send shutdown requests to the specified server until no further servers respond.

Note that, while one attach server broadcasts the SHUTDOWN command to all replicas of the service, these replicas could be running within other jobs or environments. Unlike the ENDJOB command, the ENDSERVICE command does not use the PARENT-RID to restrict the servers that carry out the shutdown request.

ENVIRONMENT Command

The ENVIRONMENT command returns information about when a server was started, as well as many of the Natural system variable values assigned within the server. You can use this information to help diagnose problems with a server and to identify the environment in which the server is running.

If the ENVIRONMENT command returns a non-blank PARENT-RID value, the service is running as a subtask. The PARENT-RID value identifies the attach server replica that hosts the server.

INITIATE Command

Use the INITIATE command to start servers. You can start as many servers as needed, to the maximum defined in the GLOBAL DEFAULTS section of the Broker attribute file. If you want to start a server or client service that uses an attach server, ensure that the attach server is running.

▶ **To initiate (start) a server:**

1   Enter "P" in `Action` for the server.

    If the system responds that the PING command was successful, there is a server with that name available on the network. If the message indicates that the server is not registered, or if you require an additional copy, continue to the next step.

2   Enter "I" in `Action` for the server.

    It can take up to 15 seconds (depending on system load) for the server to start and register with EntireX.

    Note: If the server runs as a batch job, note the name of the batch job when it is displayed as you may need it later.

3   Do one of the following:

Option	Description
Use the PING command to determine whether or not the server started properly.	■ If the PING command is successful, the server is now available. ■ If the PING command fails, press PF4 (CLog) to review the communications logs. These logs contain messages that are not communicated back to the client, such as startup errors. Use this information to determine and rectify the problem.
If this is not the first copy of the server, repeat the PING command to determine the number of replicas running (based on replica IDs).	If the server fails to start and the communication log does not reveal why, use the EntireX Control Center to determine the number of servers available or do one of the following: ■ For attach servers or servers that are not started by an attach server, it may be a problem with the JCL that is submitted. Review the batch job output and correct the JCL within the specified JCL member. ■ For servers started by an attach server, it may be a problem with the startup commands sent to Natural. Use the DEBUG command to try to identify the problem.

PING Command

Use the PING command to determine whether or not servers are running and available on the network. When a server is pinged, one of the following messages is returned:

Message	Indicates
`ETB Error 215/148 (NET: Connection Error)`	The Broker ID specified in the server definition is not currently running or responding.
`ETB Error 7/7 (server not registered)`	You have requested a valid server, but there are no servers running to satisfy the request. If this error is preceded by a message, such as "Sending command to attach server: *servername*", the attach server associated with the current server is not registered.
`ETB Error 20/216 (API: Invalid BROKER-ID)`	The Broker ID specified for the server is not valid.
`PING of server server name was successful`	At least one copy of the server that was pinged is running and available.
`ETB Error 74/74 (Wait timeout occurred)`	Either all servers are busy or a server has ended without properly de-registering.

REFRESH Command

The REFRESH command can only be sent to an attach server. When an attach server is started, it determines all servers to be registered with the Attach option by reading the server records. While the server is running, you can modify the server records and alter the set of servers that are linked to the attach server. These changes are not reflected in running attach servers until an explicit REFRESH command has been sent to all such servers.

SHUTDOWN Command

The SHUTDOWN command shuts down servers. There are three ways to shut down servers. The method you choose depends on whether you want to shut down:

- A single replica
- All replicas
- All servers within a single job

Shutdown a Single Replica

To end a single server replica, use the SHUTDOWN command with the service to be ended. To shut down a specific instance of a server, also specify the replica ID.

Shutdown All Replicas of a Server

To end multiple replicas of a server, send the End Service (ES) request to the attach server used to start the server.

Shutdown All Servers in a Batch Job

To end all servers running within a certain batch job, enter the ENDJOB command in `Action` for the attach server to be shut down. Use this option if your servers execute as subtasks of an attach server.

Example of Shutting Down Servers in a Batch Job

In addition to using the Manage Servers panel, you can also send commands directly to servers using the EntireX Driver program. This section describes the commands to implement a variety of shutdown options.

The following diagram shows an example of two batch jobs:
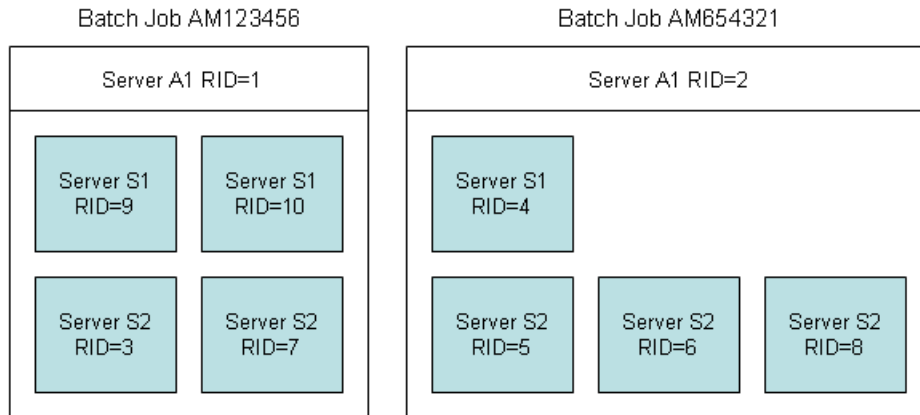
- To end any replica of server S2, send the `CMD SHUTDOWN` command to the server.

- To end server S2 replica 5, send the `CMD SHUTDOWN,RID=5` command to the server to be shut down until you receive a successful response.

- To end all replicas of server S2, send the `PROCESS SHUTDOWN SERVER S2` command to either attach server.

- To end all servers beginning with S2, send the `PROCESS SHUTDOWN SERVER S2*` command to attach server A1. Note that this command will shut down server S2 regardless of which attach server S2 is running under.

- To end all servers beginning with S2 and running batch job AM654321, send the `PROCESS SHUTDOWN SUBTASK S2*,RID=2` command to the attach server. Continue to send the command until it is received by replica 2.

- To end batch job AM123456, send the `PROCESS SHUTDOWN SUBTASK *,RID=1` command to the attach server A1. Enter the command until it is satisfied by the desired replica.

STEPLIBS Command

The STEPLIBS command returns information about the associated Natural step libraries used by the server.

SYSPROF Command

The SYSPROF command returns information about the Natural system files used by a server.

Standard Response Protocol

Most servers acknowledge or respond to standard protocol. This protocol is identified by the `MSG111` signature at the beginning of the message.

Access the Communications Logs

Access the communications logs from the Query Communications Logs panel. To access this panel, press PF4 (CLog) on the Manage Servers panel. Because each INITIATE and SHUTDOWN command generates an entry in the communications logs, this PF-key gives you quick access to the messages. Pressing PF4 (MSrvc) on the Query Communications Logs panel returns you to the Manage Servers panel.

> **Note:** For information on the communication logs, see **Queries and Logs for Servers**.

5   Using the Business Service Control Record

The business service control record provides current information about the status of your system. Using the Maintain Control Record window, you can set Entire Net-Work timeout values, enable or disable Natural Security, and specify a security exit.

> **Note:** You can also use the Security window on the client to set security privileges. For information, see Define Security for Domains.

This section covers the following topics:

Access the Maintain Control Record Window

▶ **To access the Maintain Control Record window:**

1   Enter "SA" in `Function` on the Business Service Administration Subsystem main menu.

The System Administration main menu is displayed.

2   Enter "MM" in `Function` on the System Administration main menu.

The System Administration Maintenance menu is displayed.

3   Enter "SC" in `Function` on the System Administration Maintenance menu.

The Maintain Control Record window is displayed. For example:

```
BSCT__MP  ***** Business Service Administration Subsystem *****     BSCT__11
Feb 14                     - MAINTAIN CONTROL RECORD -               8:59 PM


 Action (M)                    _

 Use Natural Security.....: _
 Security exit............: BS_USEC2
 Entire Net-Work timeout..: 60___







 Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
 confm help  retrn quit        flip  pref                               mai
```

Use the Maintain Control Record Window

Use the Maintain Control Record window to modify control data. For example, you can use this window to enable or disable Natural Security, specify a security exit, and set Entire Net-Work timeout values.

▶ **To modify the control data:**

1   Enter "M" in `Action`.

2   Change the value in the appropriate field:

- ▪ Mark `Use Natural Security` to enable Natural Security.
- ▪ Type a new value in `Security exit` to change the security exit.
- ▪ Type a new value in `Entire Net-Work timeout`.

3   Press Enter.

With Natural Security Installed

When Natural Business Services is installed, `Natural Security` is not marked. To enable Natural Security, you must mark the field.

> **Notes:**

1. Once Natural Security is enabled, all group and user maintenance and query functions are integrated into Natural Security panels and data. In addition, a query option for a user's groups and a maintenance option for User/Library Link functions are added to the menu structure.

2. For information on how the Business Service Administration subsystem is affected when Natural Security is enabled, see **Use Natural Security with Natural Business Services**.

Security Exit

By default, this field displays the routine (BS_USEC2) supplied with Natural Business Services to evaluate the security settings when requests are made to the server data. You can modify and replace the security exit used by your system.

> **Note:** For information, see **Use the Security Exit Modules**.

Entire Net-Work Timeout

When Natural Business Services is installed, the default setting for the Entire Net-Work timeout value is 60 seconds. To change the timeout setting, enter a new value in `Entire Net-Work timeout`.

> 💡 **Tip:** Adjust this value to match the lowest timeout value of all Entire Net-Work nodes running in your network.

Access the Business Service Control Record Log

All changes to the business service control record are logged. They are displayed on two panels in date and time order.

▶ **To access the business service control record log:**

1   Enter "SA" in `Function` on the Business Service Administration Subsystem main menu.

   The System Administration main menu is displayed.

2   Enter "LM" in `Function` on the System Administration main menu.

   The System Administration Log Query menu is displayed.

3   Enter "LT" in `Function` on the System Administration Log Query menu (for an example of this menu, see **System Administration Log Query Menu**).

   The System Administration Log by Time Query menu is displayed.

4   Enter "SC" in `Function` on the System Administration Log by Time Query menu.

   The first Query Business Service Control Record Logs by Time panel is displayed.

For information on the query and log panels, see **Auditing Business Services**.

6 Defining Steplibs and Domains

The business service environment includes users, application libraries, business services, and their associated modules. Users are combined into larger entities known as "groups". Application libraries, business services, and their associated modules are combined into larger entities known as "domains". Before creating an application with Natural Business Services, you must define a domain for the services. Before users can access the services, you must grant access to the business services and service methods within the domain.

This section covers the following topics:

Step 1: Define the Steplib Chain

The first step in setting up a domain is to define its steplib chain. A steplib chain identifies where your business service libraries reside on the server. To locate and execute business service modules, you must set up a steplib chain and link it to your business service domain.

When defining your steplib chain, keep the following tips in mind:

- Before adding a steplib entry, determine the database ID (DBID) and file number (FNR) of the FUSER system file you are using.

- The library in which the dispatch server is running is checked before libraries in the steplib chain; you do not have to add this library to your steplib chain.

- Both the FUSER and FNAT system libraries are automatically added to your steplib chain; you do not have to add these libraries to your steplib chain.

- If you intend to use the default DBID and FNR values for the current FUSER system file at runtime, you do not have to specify a DBID and FNR value for a library.

- Ensure that you add your FUSER file in the SYSTEM library to the steplib chain. Most business services use the server framework components supplied with Natural Construct in this file (prefixed with "CD" or "CC").

- Any components required by your generated methods, such as subprograms, copycode, or data areas, must be available in your business service library or one of its steplibs.

- If you are new to Natural Business Services, set up a sample environment. For example, set up a sample business service library and link it to your sample steplib chain. Use the same name to identify your business service library, steplib chain, and domain.

▶ **To define the steplib chain:**

1   Enter "AA" in `Function` on the Business Service Administration Subsystem main menu.

     The Application Administration main menu is displayed.

2   Enter "MM" in `Function`.

The Application Administration Maintenance menu is displayed.

3   Enter "ST" in Function.

The Maintain Steplib Table panel is displayed. For example:

```
BSSD__MP   ***** Business Service Administration Subsystem *****      BSSD__11
Feb 14                      - Maintain Steplib Table -                9:00 PM


*Action (A,B,C,D,M,N,P)     _



 Steplib Name............: _____

                          +--------------------------+

                          |   Library    DB     FNR  |

                          |   --------  -----  ------ |

                          |  1 _____  ____   ____  |

                          |  2 _____  ____   ____  |

                          |  3 _____  ____   ____  |

                          |  4 _____  ____   ____  |

                          |  5 _____  ____   ____  |

                          |  6 _____  ____   ____  |

                          |  7 _____  ____   ____  |

                          |  8 _____  ____   ____  |

                          +--------------------------+




Direct Command: _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
confm help  retrn quit        flip  pref                                  main
```

4   Add up to eight business service library chain names (`Library`), database IDs (`DB`), and file numbers (`FNR`) to the steplib chain.

5   Record the name of the steplib chain.

You will add the steplib chain to the business service domain described in the following step.

Step 2: Define the Domain

Domains are used to group related business services. You can set up the same business service in multiple domains. The services assigned to the service can be different for each domain. For example, if you have a Customer service that is used in two business services, an Accounts Receivable and a Sales business service, the Customer service in the Accounts Receivable business service probably requires different services than a Customer service in a Sales business service. Consider setting up two domains, one for each business service. Assign services to the Customer service based on the business requirements addressed by each service.

The following steps describe how to set up a domain and link it to the steplib chain described in **Step 1: Define the Steplib Chain**. By default, all business services in the domain are accessed using the same steplib chain. You can, however, override the steplib chain for each business service and service method. For information, see **Override the Steplib Chain for the Domain**.

> **Tip:** Specify a steplib chain as high in the business service architecture hierarchy as possible. This prevents you from having to specify the steplib chain in many places. If the steplib chain applies to an entire business service, place it in the appropriate domain. If the steplib chain only applies to one service, identify it in the header portion of the business service definition. In this way, only exceptions need be specified.

▶ **To define the domain:**

1   Enter "AA" in `Function` on the Business Service Administration Subsystem main menu.

     The Application Administration main menu is displayed.

2   Enter "MM" in `Function`.

     The Application Administration Maintenance menu is displayed.

3   Enter "DO" in `Function`.

The Maintain Domain Table panel is displayed. For example:

```
BSDO__MP    ***** Business Service Administration Subsystem *****      BSDO__11
Feb 14                      - Maintain Domain Table -                   9:00 PM


 Action (A,B,C,D,M,N,P)    _



 Domain Name.............: _____

 Description.............: _____

 Steplibs...............: _____ *

















Command: _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
confm help  retrn quit        flip  pref                                  main
```

4 Type "A" in `Action`.

5 Type the name of your domain in `Domain Name`.

6 Type a brief description of the domain in `Description`.

7   Type the name of your steplib chain in `Steplibs`.

8   Press Enter to add the domain.

Next, you will link the domain to user groups in **Step 3: Define Security for the Domain**.

Override the Steplib Chain for the Domain

All business services in a business service definition share the same domain. All business services within a domain are accessed using the domain's steplib chain. You can, however, override the steplib chain for each business service or method defined in your business service definition.

▶ **To override the steplib chain for the domain:**

1   Enter "AA" in `Function` on the Business Service Administration Subsystem main menu.

     The Application Administration main menu is displayed.

2   Enter "MM" in `Function`.

     The Application Administration Maintenance menu is displayed.

3   Enter "AS" in `Function`.

     The Maintain Application Service Definitions panel is displayed. For example:

```
BSIF__MP    ***** Business Service Administration Subsystem *****    BSIF__11
Feb 14,06        - Maintain Application Service Definitions -          9:01 PM


 Action (A,B,C,D,M,N,P)     _

 Domain:      _____            Service:   _____
 Version:     __ / __ / __        Type: _ * Default subprogram proxy: _____
 * Steplibs: _____

 Description:  _____

               _____

               _____

               _____

               _____

                                  Subprogram
  01            Method Name           Proxy              Steplibs *
```

```
        -------------------------------  --------  -------------------------------
    1  _____  _____  _____


    2  _____  _____  _____




  Command: _____

  Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
  confm help  retrn quit        flip  pref  bkwrd frwrd                     main
```

4   Display the business service definition for the domain you want to modify (enter "D" in `Action` and the domain name in `Domain`).

5   Type "M" in `Action`.

6   Specify the new steplib name(s) in `Steplibs`.

7   Press Enter to update the business service definition.

Step 3: Define Security for the Domain

To make your business service available to users, you must grant them security privileges. To set up security, assign users to groups. Groups identify users who require similar access privileges to your business service. You can then grant groups security privileges to your business service domain. Granting access to a domain enables users to access the services and methods within the domain.

> **Tip:** You can postpone this task until after you have created and tested your business service. At that time, you can better determine which security privileges should be granted to each group.

For each group granted access to a domain, you can further define security privileges by granting access to selected services and methods. For example, assume you have a business service domain called "Payroll" containing all of the services and methods required for your organization's payroll business service. Two types of users require access to the payroll business service: managers and data entry personnel. Managers require access to the entire business service, while data entry personnel only require access to input hours, vacation time, sick time, and so on. You can set up one group for the managers and one for the data entry personnel. The Manager group is given access to all services and methods in the Payroll domain and the Data Entry group is only given access to those services and methods required to do their job.

> **Notes:**

---

1. For information on security, see **Defining Users and Security Groups**.

2. For information on defining security for groups and domains, see **Setting Business Service Security Options**.

Change Security Privileges for the DEVS Group

By default, the DEVS group has access to all the supplied business services used by developers. Since security is applied automatically to all new domains, only users in the DEVS group are permitted to use these services (to test a new service, for example). If your organization uses a different group name for development purposes, you can modify the BS_USEDD subprogram in the SYSBIZ library and change the group name.

7   Defining Users and Security Groups

This section describes how to define users and how to define groups, which represent a collection of users and provide security options for domains, business services, and/or methods. These definitions allow you to set up a secure environment for your business services. With one setting, you can grant or deny a group of users access to one or more domains, or set privileges for individual services and methods.

▶ **To set up a secure environment for your business services:**

1   Define groups to represent collections of users.

2   Define users in the Business Service Administration subsystem.

3   Associate users with the appropriate groups.

4   Define domains to represent collections of services associated with related business functions.

> **Note:** You can define users and groups in either Natural Business Services or Natural Security. For more information , see **Setting Business Service Security Options**.

This section covers the following topics:

Introduction

The following table describes the components to create a secure environment for a business service:

Component	Description
Group	Represents multiple users with the same need and authority to access particular business services and their methods. For example, an organization may have separate groups for Quality Assurance, Sales, and Administration.
User	Represents individual users in the Business Service Administration subsystem. User profiles define the groups to which each user belongs.
Domains, services, and methods	Domains represent a collection of several business services related to a business function. Each business service has access to one or more methods. **Note:** Security at the domain, service, and/or method level is only available with Natural Business Services.

The following diagram shows the relationships between a user and the associated group, domains, services, and methods:
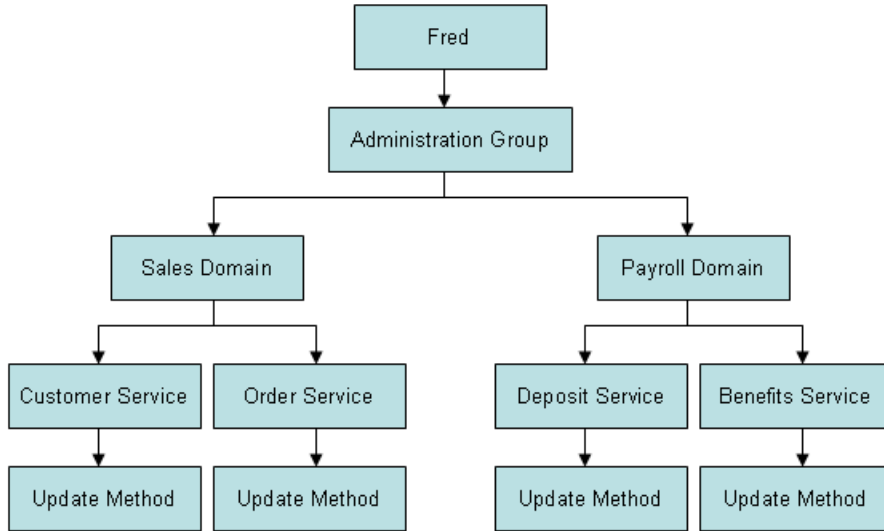
Because the user, Fred, is a member of the Administration group, he potentially has access to all domains, services, and service methods associated with that group. But in this example, Fred has only been granted access to two domains, four services, and one method.

When the CSRLOAD program is run during installation, the following groups and sample users are loaded into the Business Service repository:

Group	Sample User	Function	Security Access
ADMIN	PWRUSR	Administration activities, such as: ■ Auditing business service activities ■ Deploying services ■ Applying security ■ Developing code ■ Invoking the DEMO domain services	Full access to the ADMIN, INTERNAL, and DEMO domains.
DEVS	PWRUSR DEV*nn*	Development activities, such as: ■ Developing new business services in Natural ■ Creating services in Visual Studio and/or Java	Full access to the INTERNAL and DEMO domains, as well as access to the following methods in the ADMIN domain: ■ GET ■ BrowseDomain

Group	Sample User	Function	Security Access
		Note: This group is restricted to development activities; it cannot access security data and cannot deploy, audit, or maintain domains.	■ BrowseServices ■ FindByDomainName ■ FindBySteplibName
USERS	PWRUSR DEV*nn* BUSER	Execute Web services developed using the services supplied in the DEMO domain.	Full access to the DEMO domain.
DEMOUSRS	USER01 USER02 USER03 USER04 USER05	Training activities. **Note:** 1. Trainees do not automatically belong to the DEMOUSRS group; you must add each trainee to the group. 2. To allow a trainee to run or test a newly created business service in a training domain, the trainee must be a member of the DEMOUSRS group. 3. To allow trainees to develop and run services in the DEMO domain, they also belong to the DEVS and USERS groups.	Full access to the DEMO01-DEMO05 domains. **Tip:** When you assign a user to the DEMOUSRS group, the user will have access to all DEMO*nn* domains. To restrict the user to one domain, you can create a group that only has access to that domain and then assign the user to that group.

> **Note:** The PWRUSR user also has access to the CST and VBADDIN domains (Visual Basic 6 only).

If Natural Security (NSC) is being used, you must add the following groups to Natural Security and assign the appropriate user IDs to each group:

■ ADMIN

■ DEVS

■ USERS

Users in these groups must belong to the appropriate groups. For example:

■ To perform administration functions (for example, to view audit trails), the user must belong to the ADMIN group.

■ To develop a Natural business service, the user must belong to the DEVS group.

■ To use the demo application, the user must belong to the USERS group.

The remainder of this section describes how to define groups and users, using only Natural Security or Natural Business Services security.

> **Note:** For information on the queries and logs available to view information about groups and users, see **Auditing Business Services**.

Define Groups

All users of a business service that employs servers must be known to the Business Service Administration subsystem through their associations with groups. To access the business service services, a user must be a member of at least one group that is linked to a domain containing the service. A user can belong to a maximum of 10 groups.

Groups are used to collect users into larger entities so that security settings can be defined and checked at a higher level. This eliminates the need to define security settings for each user.

> **Tip:** To define security privileges for a single user, create a group containing that user alone.

Natural Business Services allows users and groups to be defined either within the business service system files or by linking to the Natural Security user and group definitions.

> **Note:** For more information, see **Use Natural Security with Natural Business Services**.

This section covers the following topics:

- Under Natural Business Services Security
- Under Natural Security

Under Natural Business Services Security

Use the Maintain Group Table panel in the Business Service Administration subsystem to define groups. Each user can belong to one or more of these groups.

▶ **To define a group under Natural Business Services security:**

1   Enter "SA" in `Function` on the Business Service Administration Subsystem main menu.

    The System Administration main menu is displayed.

2   Enter "MM" in `Function` on the System Administration main menu.

    The System Administration Maintenance menu is displayed.

3   Enter "GR" in `Function` on the System Administration Maintenance menu.

The Maintain Group Table panel is displayed. For example:

```
BSGR__MP   ***** Business Service Administration Subsystem *****      BSGR__11
Feb 14                       - Maintain Group Table -                  9:02 PM


 Action (A,B,C,D,M,N,P)    _



  Group ID................: _____

  Group Name..............: _____


















Command: _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
confm help  retrn quit       flip  pref                                  main
```

4   Type "A" in `Action`.

5   Type the group identifier in `Group ID`.

6   Type the name of the group in `Group Name`.

7   Press Enter to confirm the information.

After adding all required groups, you can use the group identifiers when defining users.

Under Natural Security

When Natural Security is enabled in the Business Service Administration subsystem, use the Maintain Group Profile window instead of the Maintain Group Table panel supplied with Natural Business Services to define groups.

▶ **To define a group under Natural Security:**

1   Enter "SA" in `Function` on the Business Service Administration Subsystem main menu.

     The System Administration main menu is displayed.

2   Enter "MM" in `Function` on the System Administration main menu.

     The System Administration Maintenance menu is displayed.

3   Enter "GR" in `Function` on the System Administration Maintenance menu.

     The Maintain Group Profile window is displayed. For example:

```
          Maintain Group Profile (Natural Security)


   Function ..........................  __
   User ID ...........................  _____
   User Type .........................  G
   Library ID or copied from user .....  _____



              Press any PF-key to stop
```

4   Type "AD" in `Function`.

     AD is the code for the Add action. Other codes for this field are:

Code	Description
CO	Performs the Copy action.
DI	Performs the Display action.
MO	Performs the Modify action.

5   Type the name of the new group in `User ID`.

     Each group ID must be unique among all user IDs and library IDs defined to Natural Security.

6   Type "G" in `User Type`.

7   Press Enter.

The Add User panel is displayed. For example:

```
 15:35:15                    *** NATURAL SECURITY ***              2006-03-06
                                - Add User -



                                         Modified ..            by

 User ID ...... NEWUSER

 User Name .... _____

 User Type .... G (Group)



 Members          Libraries        Batch User ID ..... _____

 --------      -------------------

 _____         Default .. _____       Time   Differential . + __ h  30 min

 _____                                    Zone    _____
 _____

 _____                                    Language .......... _0

 _____                                    Logon recorded .... N




 No. members




 Additional Options ... N



 Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
       Help  PrevM Exit  AddOp      Flip                               Canc
```

8   Use this panel to add the group.

A group is distinguished from an individual user by its `User Type` value: G.

> **Notes:**

1. You must define all required group and domain combinations using the Maintain Service Security by Group panel. For information, see **Add a Domain/Group Combination**.

2. For more information on using the Add User panel, refer to the Natural Security documentation for mainframes.

Define Users

Business service administrators define users so that their passwords can be checked when they attempt to perform actions in the business services supported by the Business Service Administration subsystem.

- If Natural Security is not enabled, use the Maintain User Table panel to identify each user, specify the user's debug library and file, specify the user's preferred language, and list the groups to which the user belongs.

- If Natural Security is enabled, use the Maintain User Profile window and Add User panel to define users. To associate users with groups, use the Maintain User's Groups window. You can also link users to protected libraries using the Library Maintenance/Link Users window. If you are using Natural Security, the user's debug library and file name default to the user ID. By default, the preferred language is English.

Under Natural Business Services Security

▶ **To define a user using Natural Business Services security:**

1   Enter "SA" in `Function` on the Business Service Administration Subsystem main menu.

    The System Administration main menu is displayed.

2   Enter "MM" in `Function`.

    The System Administration Maintenance menu is displayed.

3   Enter "US" in `Function`.

The Maintain User Table panel is displayed. For example:

```
BSUS__MP    ***** Business Service Administration Subsystem *****    BSUS__11
Feb 14,06                    - Maintain User Table -                    9:08 PM


 Action (A,B,C,D,M,N,P)     _



 Userid/Passw must be in upper case and <= 8 chars for Mainframe and/or NSC___
 User ID.................: _____

 Password...............:

 Name...................: _____


 Debug Library..........: _____

 Debug Filename.........: _ ('T'imestamp; 'U'ser ID)



 Preferred Language......: __

 Groups...: _____  _____
            _____  _____
            _____  _____
            _____  _____
            _____  _____
Direct Command: _____

Add        Browse     Clear      Display    Modify     Next       Purge

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
confm help  retrn quit        flip  pref                                  main
```

Use this panel to specify a user's debug library and file. The `Debug Library` value defaults to your user ID.

> **Note:** Using Trace-Option(1), you can store data transmitted between the client and server in the debug file.

4   Type the code for how the debug file is named in `Debug Filename`.

Valid codes are:

Code	Option	Description
T	Timestamp	Current time value is the file name. This allows multiple copies of debug data to exist; later copies do not overwrite earlier ones. Each file is assigned a name of "T" followed by a time value of `HHMMSST`. For example, files generated at 12:23:54.7 and 18:12:51.9 have the names "T1223547" and "T1812519", respectively.
U	User	User ID is the file name. This option is the default. Each request from the client overwrites the contents of the debug file.

> **Note:** When a user invokes a subprogram proxy from the Application Administration main menu, the default values for the debug library and file are taken from the settings defined on the Maintain User Table panel.

5   Type the name of at least one group in `Groups`.

Each user must be associated with at least one group. If a user does not belong to a group, Natural Business Services rejects all access requests made by the user. When defining security settings, associate groups and domains to specify which users have access to which services. Each user must belong a group that has been associated with the domain containing the services the user needs to use. You can specify group(s) when adding a new user or modifying the profile of an existing user.

6   Press Enter to define the user.

Under Natural Security

When Natural Security is enabled in the Business Service Administration subsystem, use the Maintain User Profile window instead of the Maintain User Table panel supplied with Natural Business Services to define users. To associate users with groups, use the Maintain User's Groups window. You can also link users to protected libraries using the Library Maintenance / Link Users window.

▶ **To define a user using Natural Security:**

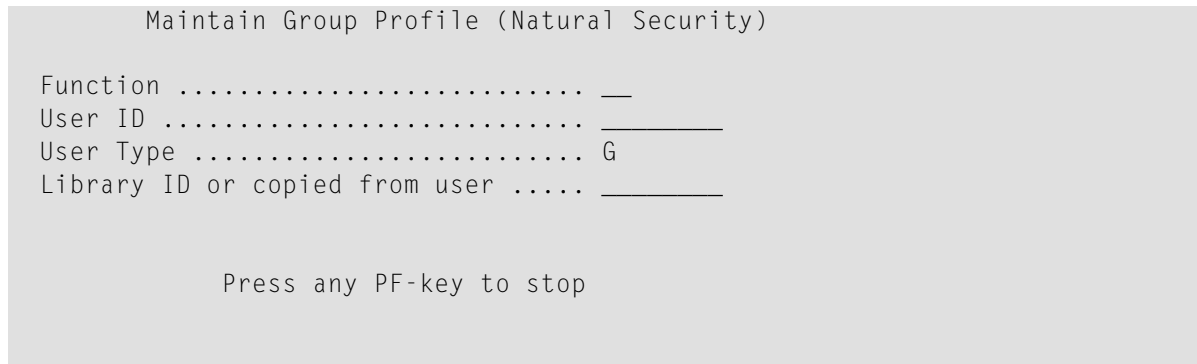1   Enter "SA" in `Function` on the Business Service Administration Subsystem main menu.

The System Administration main menu is displayed.

2   Enter "MM" in `Function`.

The System Administration Maintenance menu is displayed.

3   Enter "US" in `Function`.

The Maintain User Profile window is displayed. For example:
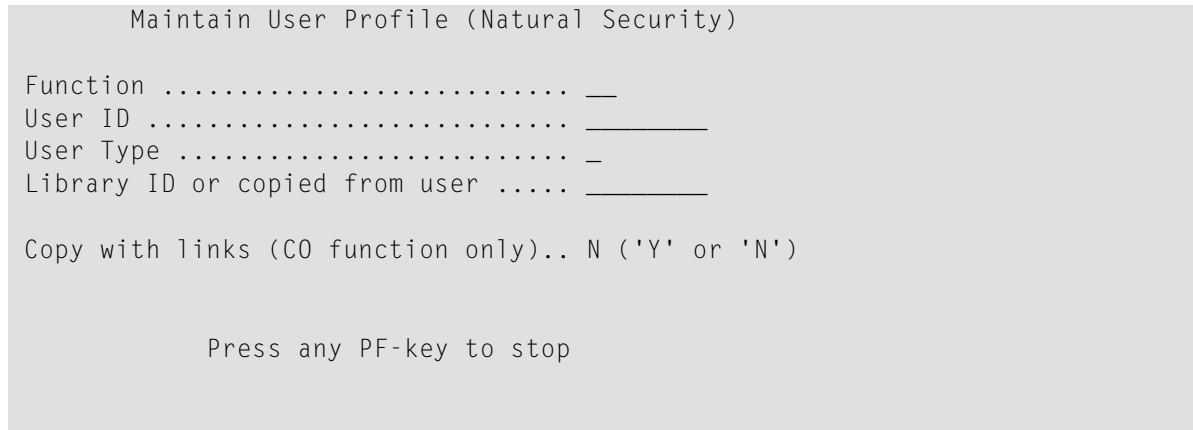
```
        Maintain User Profile (Natural Security)


Function .......................... __
User ID ........................... _____
User Type ......................... _
Library ID or copied from user ..... _____

Copy with links (CO function only).. N ('Y' or 'N')



          Press any PF-key to stop

```

4   Type "AD" in `Function`.

AD is the code for the Add action. Other codes for this field are:

Code	Description
CO	Performs the Copy action.
DI	Performs the Display action.
MO	Performs the Modify action.

5   Type the name of the new user in `User ID`.

Each user ID must be unique among all user IDs and library IDs defined to Natural Security.

6   Type the code for the type of user in `User Type`.

Valid codes for this field are:

Code	Description
A	User is an administrator.
M	User is a member of a group.
P	User is a person who is not an administrator or a member of a group.

7   Press Enter.

The Add User panel is displayed. For example:

```
14:29:10                     *** Natural Security ***                    2006-03-13
                          - Add User -

                                                    Modified .. 98-02-21 by DEVRE
User ID ...... NEWUSER
User Name ....  _____
User Type .... P (A=Administrator, P=Person, M=Member)

Privil. Groups            Libraries                      Password
-------------         ------------------          ----------------------
_____             Default .. _____         New Password
_____             Last ..... _____         Change after ___ days
_____
_____                  ETID                   Batch User ID ..... _____
_____             ------------------          Language .......... _0
                      Default .. _9993  G         Private Library ... Y
No. groups 3          Last ..... _____         Logon recorded .... N


Additional Options ... N

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help            Exit  AddOp        Flip                              Canc
```

8   Use this panel to specify information about the user.

> **Note:** For a description of this panel, refer to the Natural Security documentation for mainframes.

Associate Users with Groups

Each user must be associated with at least one group. If a user does not belong to a group, Natural Business Services rejects all access requests made by the user. When defining security settings, associate groups and domains to specify which users have access to which services. Each user must belong to a group that has been associated with the domain containing the services the user needs to use.

▶ **To associate a user with a group:**

1   Enter "SA" in `Function` on the Business Service Administration Subsystem main menu.

     The System Administration main menu is displayed.

2   Enter "MM" in `Function`.

     The System Administration Maintenance menu is displayed.

3   Enter "UG" in `Function`.

The Maintain User's Groups window is displayed. For example:

```
        Maintain User's Groups (Natural Security)

 Group ID ......................... _____
 ADd / DElete / LIst members .......
 Start value for list function ...... _____

 Members

  1 _____   2 _____   3 _____   4 _____
  5 _____   6 _____   7 _____   8 _____
  9 _____  10 _____  11 _____  12 _____
 13 _____  14 _____  15 _____  16 _____
 17 _____  18 _____  19 _____  20 _____
 21 _____  22 _____  23 _____  24 _____
 25 _____  26 _____  27 _____  28 _____
 29 _____  30 _____

            Press any PF-key to stop
```
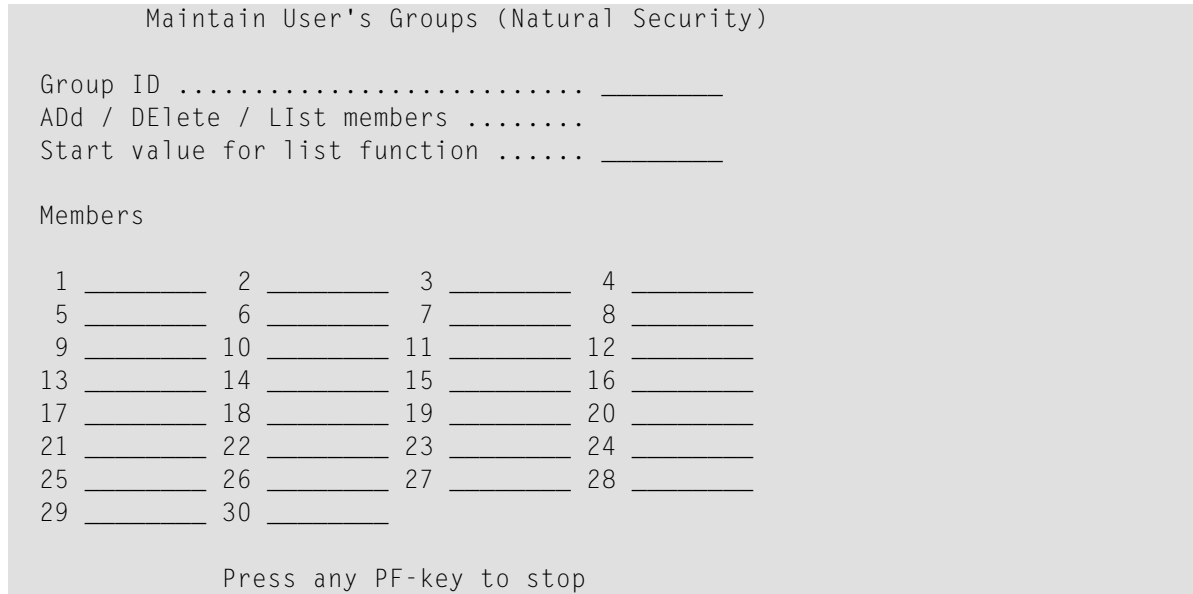
4   Type the group name in `Group ID`.

5   Type "LI" in `ADd / DElete / LIst members`.

6   Press Enter.

The group's members are listed in the Members columns.

7   Type "AD" in `ADd / DElete / LIst members`.

8   Type the user ID you want to add in an empty field in `Members`.

9   Press Enter.

The user is added to the group.

Link Users to Libraries

Use the library Link function to associate users with protected libraries. The Business Service Administration subsystem uses the link definitions found in Natural Security to determine whether or not a user can access a Natural library. Each library must be linked to the user in Natural Security. The libraries include:

■ Each protected library in a steplib

■ The current execution library, if protected

■ The SYSTEM (or *STEPLIB library), if protected

▶ **To link a user to a protected library:**

1   Enter "SA" in `Function` on the Business Service Administration Subsystem main menu.

    The System Administration main menu is displayed.

2   Enter "MM" in `Function`.

    The System Administration Maintenance menu is displayed.

3   Enter "LI" in `Function`.

    The Library Maintenance/Link Users window is displayed. For example:

```
   Library Maintenance / Link Users (Natural Security)


 Function ........................... LK
 Library ID ......................... _____
 Library/private library ............ L

 User ID (for link functions)........ _____

             Press any PF-key to stop
```

4   Type the library name in `Library ID`.

5   Type the user ID in `User ID`.

6   Press Enter.

    The link is created.

8   Setting Business Service Security Options

This section describes how to set Natural Business Services security options for business services. These options control access at the domain, service, and method level. When the client requests access to data on the server, the dispatch server invokes the security server, which executes a Natural routine in the security exit. The routine analyzes the security settings to determine whether the user is allowed to access the data.

■ If the user has permission, the call proceeds.

■ If the user does not have permission, a message is returned to the client and the call ends.

This section describes how to define the security settings and how to use the supplied security exits to interpret the security settings. The following topics are covered:

Introduction

Natural Business Services security allows you to define a group's access privileges at multiple levels: domain, business service, and method. However, you do not have to define security settings for every service and method. By default, security privileges set at one level will automatically apply to all lower levels in the hierarchy. In addition, you can define different settings for individual items at a lower level. As well, Natural Business Services provides security exit modules (BS_USEC*) you can modify to evaluate the security settings as needed. Security privileges can be as general or as specific as needed.

At any time, you can disable current access privileges. This allows you to perform maintenance without changing the security settings. The disabled status is removed by applying the enabled status, which returns settings to their prior state.

The version number in a business service definition is ignored in security definitions. When adding a new version of a service that is already defined to security, the new version is automatically assigned the same privileges as older versions.

Server Security Overview

When using Natural Business Services (NBS), it is important to understand how information on the server is secured. For example, when working in Natural for Windows, SPoD is the only middleware option and all security checking is performed through the Natural Development Server (NDV). EntireX is not used, not even by the NBS attach, dispatch, and security servers. The Business Service repository, however, handles user authorization.

The following diagram illustrates the different security options on the server:
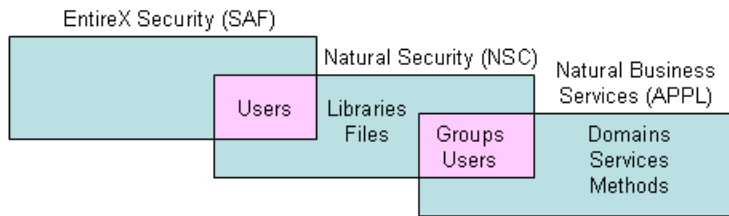
The pink areas indicate overlapping options. For example, users can be authenticated in EntireX, Natural Security, or Natural Business Services, but groups can only be authenticated in Natural Security or Natural Business Services security.

Security options on the server are:

■ NONE

 No security; program names and libraries are checked.

■ APPL

 Natural Business Services authenticates users and checks authorizations and application-level security.

■ NSC

 Natural Security authenticates users and checks authorizations.

■ SAF

 EntireX authenticates users and checks authorizations.

■ SAF-APPL or SAF-NSC

 Hybrid security server is used (for information, see **Use a Hybrid Security Server with Natural Business Services**).

With the appropriate security permissions, you can monitor breaches of security using the logs stored in the repository.

Note: For information on monitoring security exceptions in the Natural Business Services plug-in for Eclipse, see Audit Security Exceptions for a Connection. For information on monitoring security exceptions in the Natural Business Services plug-in for Natural, see Audit Security Exceptions for a Connection.

This section covers the following topics:

   ■ Natural Business Services Authorization

- User, Group, and Library Maintenance
- SPoD (Natural for Windows) Connectivity
- EntireX (SAF) Connectivity
- Web Application and Web Service Security in Visual Studio Add-in
- Server Security Tips and Techniques

Natural Business Services Authorization

The Natural for Windows development environment is always secured and authorization privileges are based on the settings defined in the Business Service repository. It is important to understand the domain/service/method security hierarchy and how the interpretation of this hierarchy can change based on settings in the security exit modules.

Because of the various ways to interpret the security privileges, and for better performance when determining the authorization of a particular request, Natural Business Services uses a security cache system. Once a user has been authorized for a particular request, the permission is placed in the security cache.

Note: For information on this process, see **Use the Cache Synchronization Exit Module**.

User, Group, and Library Maintenance

To maintain users, groups, and libraries, the Business Service Administration subsystem provides a user interface into Natural Security. Alternately, you can maintain this information in Natural Security.

Notes:

1. For information on the user interface to Natural Security, see **System Administration Menus with Natural Security**.

2. For information on maintaining the information in Natural Security, see **Defining Users and Security Groups**.

SPoD (Natural for Windows) Connectivity

This connectivity option is required for Natural for Windows and is an option for the Visual Studio add-in. (It is not currently available for the Ellipse plug-in). In Natural for Windows, security authentication for Natural Business Services uses the data in either the Business Service repository or in Natural for Windows, depending on the settings on the business service control record.

Note: For information on the control record, see **Using the Business Service Control Record**.

EntireX (SAF) Connectivity

This connectivity option is required for the Ellipse plug-in and is an option for the Visual Studio add-in. (It is not available in Natural for Windows.) EntireX balances loads, which makes it an ideal choice for production (runtime) applications. For Natural Business Services, this option includes attach, dispatch, and security servers and an Attach Manager. In addition, the EntireX compression, encryption, and security options are available.

You do not have to use EntireX security when using the EntireX middleware. The security option is determined by the dispatch server definition (not the business service control record). For example, to use Natural Security with EntireX connectivity, the security mode setting for the dispatcher must be NSC. Depending on the security mode setting for the dispatch server, a user can be authenticated using EntireX, Natural Security, and/or Natural Business Services security.

📄 **Notes:**

1. On the server, EntireX security checks the user ID and password. On the mainframe, this only happens if the security mode for the dispatch server is SAF, SAF-APPL, or SAF-NSC. In addition, the client configuration must use a Kernel security mode of "Auto" or "Yes".

2. For information on defining servers, see **Define Server Settings and Program Parameters**.

3. For information on using EntireX security, see **Use EntireX Security with Natural Business Services**.

EntireX Server Maintenance

When setting up a new environment, it is sometimes difficult to define and manage the EntireX servers. Natural Business Services provides the CSRLOAD utility in the SYSBIZ library to simplify this process. Use this utility to load the business services and security settings, as well as the customized servers (attach, dispatch, and security). Since the settings get more complicated for a secure environment, we recommend that you use CSRLOAD to load servers that will be using Natural Security.

▶ **To load the secure servers:**

1 Invoke the CSRLOAD utility from the SYSBIZ library.

2 Type the broker ID in Broker ID.

3 Type a prefix or suffix in the Server qualifiers fields.

This value will identify the secure servers within Natural Business Services. If you type "NSC" in the Prefix field, for example, the server names will be NSC-Attach, NSC-Dispatch, and NSC-Security.

4 Type the server class in Server Class.

This value is found in the EntireX Broker attribute file.

5   Type the name of the JCL text member in JCL Text Member.

The JCL text member must be modified to handle Natural Security. For instructions on modifying the text member, create your own copy of BATCHJCL in the SYSBIZ library and read the comments.

6   For production environments, type "Y" in Subtask Support.

If subtask support is specified, the dispatch and security servers run under the attach server in Natural. This is less resource-intensive, but makes it more difficult to find problems. If subtask support is not specified, all servers are set up as independent batch jobs.

> **Tip:** Tip: For debugging purposes, it may be easier to not use this option (i.e., type "N" in Subtask Support).

7   Ensure that the database ID and file number for the batch Natural profile are specified in Profile FNAT DBID and Profile FNAT FNR.

The Natural profile is used to start the servers.

8   De-select all other options in the Mark default records to be loaded section.

For example:

```
This process will load the current Business Files. Make sure your LFILES
are set correctly. This program also clears the source area and generates
entries required for the BROKER Attribute File. These must be copied manually.

Enter default values to be used to establish server records...
  Broker ID ........ BKRnnn_____
  Server qualifiers Prefix: _____  Suffix: _____
  Server Class ..... BUSINESS_____ Unicode (Y/N): N
  JCL Text Member .. BATCHTXT
  Transaction ...... NATBAT__
  Subtask Support .. Y
  Profile FNAT DBID  _13000
  Profile FNAT FNR . __1300
  Profile .......... SYSBIZ__

Natural Security Settings...
  Servers under NSC  _
  User id .......... _____
  Password ........

Mark default records to be loaded...
  _ Users              _ Groups            _ Domains              _ Steplibs
  X Servers            _ Business Services _ Security Links
```

9   Press Enter.

> **Note:** If you select Servers under NSC, the additional parameters for Natural will correctly set up the stack on the assumption that the specified user ID and password are also in Natural Security.

To verify what was loaded, access the Manage Servers panel. For information on accessing this panel, see **Manage Servers**.

Web Application and Web Service Security in Visual Studio Add-in

Web applications that were generated from the Visual Studio add-in use Microsoft's IIS technology and the Web Service Engine (WSE) to provide application security and use EntireX as the middleware component. The combination of these components provides the following security modes for passing a password across the network: None, Token, or Password.

> **Note:** For information, see Build a Web Application.

Server Security Tips and Techniques

This section contains tips and techniques you can use to help troubleshoot security problems. The following topics are covered:

- Test EntireX Connection
- Correct Invalid User ID or Password Error

Test EntireX Connection

You can use the Manage Servers panel in the Business Service Administration subsystem to initiate and ping each server. This is highly recommended when setting up the servers for the first time. For information, see **Manage Servers**.

> **Note:** You can also use a GUI application called the Server Monitor to manage servers. For information on using the Server Monitor, see NBS Server Monitor.

When an attach server is initiated from the Manage Servers panel, a batch job is started and the name of the batch job is displayed in a message on the panel. If you encounter a problem, view the batch job (and/or the batch jobs for the dispatch and security servers, if they are running as batch jobs).

If the servers do not respond, confirm that the JCL and startup parameters are correct. The startup parameters in the JCL template should look like the following example:

```
SYSBIZ,&USER-ID,&PASSWORD
SPSSTART
```

The startup parameters should not look like the following example:

```
LOGON SYSBIZ
SPSSTART
```

Next, confirm that the server start parameters in the Natural Business Services server definitions are correct for a secure environment. For example:

```
NATPARM='FNAT=(1,1),PROFILE=SYSBIZ
AUTO=OFF STACK=(LOGON SYSBIZ,*USER-ID,*PASSWORD;START)'
```

> **Note:** If you use *USER-ID and *PASSWORD, the user ID and password associated with the server definition should be the same as those associated with the operating system. They should also be defined in Natural Security. In addition, the user ID associated with the security server must have access to the SYSSEC library.

Once the attach, dispatch, and security servers can be successfully pinged, test the client configuration from the Natural Business Services plug-in or add-in. Once a configuration has been set up, you can use the Test button on the Configuration Utility panel to test the connection.

> **Note:** If a dispatch server is not required, it shuts down and is only initiated for a request. This may cause the dispatch server to fail the test. If this happens, either initiate the dispatcher from the server or invoke one of the wizards (which will force a request to the dispatcher and initiate the dispatch server).

Correct Invalid User ID or Password Error

An invalid user ID or password error can be the result of many things. Use the checklist in one of the following sections to determine the problem.

SPoD (Natural for Windows) Connection

If the problem occurs from a SPoD (Natural for Windows) connection:

1. Verify that the user ID and password were entered correctly.

   The most common reason for this error is that the wrong user ID or password was entered or that the wrong case was used (for example, mixed case when it should be upper case).

2. Verify that the user ID and password entered are the same as those used to map to the Natural remote environment.

3. Determine where on the server the user ID and password are being checked.

   From a SPoD connection, there are two options: Natural Security or the Business Service repository. Confirm that the control record in the Business Service Administration subsystem specifies the correct security option (direct command: "MENU SA MM SC"). For example:

   ■ If the business service control record has the Use Natural Security option marked, Natural Business Services will verify the user ID in Natural Security. Confirm that the user ID and password entered are the same as those defined in Natural Security.

- If the Use Natural Security option is not marked, Natural Business Services will verify the user ID and password in the Business Service repository. This security information is maintained in the Business Service Administration subsystem on the server (direct command: "MENU SA MM US").

4. If the problem still exists, confirm that the files you are using in Natural for Windows are the same as the files on the server. For example:

   - If Natural Security is being used, confirm that the FSEC file in SPoD is the same as the FSEC file on the server (you can use SYSPROF to do this).

   - If Natural Security is not being used, confirm that LFILE 135 and 136 in SPoD are the same as those on the server.

EntireX Connection

If the problem occurs from an EntireX connection:

1. Determine which user ID and password are being used and where the security information is verified.

   From the Visual Studio add-in or the Eclipse plug-in, the user is prompted to logon when a connection is made. Note the logon connection, which is determined by the default Natural connection.

2. Confirm that the secure server is marked in the EntireX configuration. (If you do not do this, the password is never sent to the server.)

3. Confirm that the attach, dispatch, and security servers can be initiated and pinged from the Manage Servers panel.

   Tip: Although you do not have to initiate the dispatch or security servers, doing so will verify that the user IDs are working for the appropriate Natural batch job. For example, if the user ID associated with the batch job does not have access to the SYSSEC library, the security server will not be initiated.

4. Determine whether the LFILE (for APPL security) or FSEC (for Natural Security) files are appropriate.

   To do this, access the Manage Servers panel and enter "SY" in the Action field for each server. If the files are not correct, define the correct location in the Natural profiles or JCL member and then initiate and shutdown the servers.

5. If the servers can be pinged and the user ID is still invalid from the client, determine which SECURITY-MODE option is defined on the second Maintain Servers panel for the dispatch server. Regardless of which setting is defined on the business service control record, the dispatch server will verify the information based on the security mode specified on this panel. Different security options can be defined for a dispatch server, since there can be more than one (for example, one dispatch server may be secured and one may not be secured). For example:

■ If SECURITY-MODE=NONE or SECURITY-MODE=APPL, the user ID is defined in the Business Service Administration subsystem.

■ If SECURITY-MODE=SAF, the user ID is defined in SAF (for example, your server operating system).

■ If SECURITY-MODE=NSC, the user ID is defined in Natural Security.

> **Notes:**

1. For information on the Manage Servers panel, see **Manage Servers**.

2. For information on the SECURITY-MODE option on the second Maintain Servers panel, see **Define Server Settings and Program Parameters**.

3. The control record provides user interfaces into the appropriate repositories, either Natural Security or Natural Business Services. SPoD uses this information to determine which security option to use to verify the user ID.

Define Security Settings

To define security settings, use the Business Service repository in Natural for Windows. For information, see Define Security for Domains. Although you can define the security settings on the server, it is simpler to use the Natural Business Services plug-in in Natural for Windows.

This section describes how to define security settings on the server. The following topics are covered:

■ Access the Maintain Service Security Panels
■ Add a Domain/Group Combination
■ Purge a Domain/Group Combination
■ Set Security Options
■ Security Statuses

Access the Maintain Service Security Panels

You can view security information in alphanumerical order either by group or by domain. The following steps describe how to access both options.

▶ **To access the Maintain Service Security panels:**

1 Enter "AA" in `Function` on the Business Service Administration Subsystem main menu.

The Application Administration main menu is displayed.

2 Enter "MM" in `Function`.

The Application Administration Maintenance menu is displayed.

3   Enter the code for the panel you want to display.

Valid codes are:

Code	Panel	Description
OG	Maintain Service Security by Group panel	Displays groups in alphanumeric order.
OD	Maintain Service Security by Domain panel	Displays domains in alphanumeric order.

Add a Domain/Group Combination

When you add a new group in the Business Service Administration subsystem, the group does not automatically appear on the Maintain Service Security panel. You must first add a domain/group combination on either the Maintain Service Security by Group or Maintain Service Security by Domain panel. The following steps describe how to add a combination on the Maintain Service by Group panel, but the procedure is the same on the Maintain Service by Domain panel.

▶ **To add a group and domain combination:**

1   Type the name of the group in `Group` on the Maintain Service by Group panel.

2   Type the name of the domain in `Domain`.

3   Press PF4 (Add).

The new domain/group combination is added to the list on the panel.

Adding a domain/group combination does not automatically give the group access to the domain. You must explicitly set privileges using one of the actions available on the panel.

🗎   **Notes:**

1. If you add a new service to an existing domain, it automatically inherits the security privileges of the domain/group combination to which it belongs. If you add a new method, it inherits the same security settings as the service to which it belongs.

2. For information on the security options, see **Set Security Options**.

3. For information on assigning security statuses, see **Security Statuses**.

Purge a Domain/Group Combination

The Purge action is only available at the domain/group level. It removes all security statuses for the domain/group combination and its associated services and methods. While deleting the security definitions, it also deletes all related entries from the security cache. Once all security definitions have been removed, the Purge action also removes the domain/group combination.

Set Security Options

This section describes how to set security options at the domain/group, service, and method levels. Although the panels shown are the Maintain Service Security by Group panels, the descriptions also apply to the Maintain Service Security by Domain panels.

> **Note:** For information on the security statuses you can assign to domain/group combinations, services, and methods (such as grant or disable), see **Security Statuses**.

Domain/Group Level

You can define settings for any combination of domain and group. If a setting is defined only at the domain/group level, the group has the same privilege for all services and methods within the domain.

▶ **To set security options for a domain/group combination:**

1   Enter the appropriate code in `Act` for the group.

    The Maintain Service Security by Group panel is displayed.

2   Press Enter to display the available groups.

    For example:

```
BSOSG_MP   ***** Business Service Administration Subsystem *****      BSOSG_11
Feb 15,06            - Maintain Service Security by Group -           10:29 PM


                                               Current   Previous

Act              Group                Domain   Access    Access

---  --------------------------------- -------- -------- --------

  _    ADMIN                            ADMIN    Revoked

  _    ADMIN                            CST      Granted

  _    ADMIN                            DEMO     Revoked
```

```
_   CST                                    ADMIN    Granted

_   CST                                    CST      Granted

_   CST                                    DREW     Granted

_   DEMO                                   ADMIN    Granted

_   DEMO                                   TEST1    Granted

_   DEMO                                   BUSINESS Granted

_   DEMO                                   DEMO     Granted

_   DEMO                                   TEST2    Granted

_   DEMO                                   TEST3

_   TESTGROUP                              ADMIN

Group: _____ *  Domain: _____ *

Direct command...: _____
(PF4=Add)       Clear      Disable    Enable      Grant      Purge      Service
(PF5=flip)
```

> **Tip:** Use the Clear action to remove security settings for a domain/group combination. This action also clears the domain/group's settings from the security cache.

> **Note:** For more information on setting security options for a domain/group, see **Add a Domain/Group Combination**.

Business Service Level

Use the second level in the security hierarchy to allow the group access to specific services within a domain or to override settings defined at the domain/group level.

▶ **To set security options at the service level:**

1   Enter "O" in Act for the domain/group combination.

    The Maintain Service Security by Domain window is displayed.

2   Press Enter to display the available groups and domains:

    For example:

```
BSOSD_MP   ***** Business Service Administration Subsystem *****      BSOSD_11
Feb 15,06              - Maintain Service Security by Domain -          10:27 PM


                                                      Current   Previous

Act       Domain                   Group              Access    Access

---       --------  ---------------------------------- -------- --------

  _       ADMIN     ADMIN                              Revoked

  _       ADMIN     CST                                Granted

  _       ADMIN     DEMO                               Granted

  _       ADMIN     TESTGROUP
  _       ADMIN2    DEMO                               Granted

  _       BUSINESS  DEMO                               Granted

  _       BUSINESS  RESTRICT

  _       CST       ADMIN                              Granted

  _       CST       CST                                Granted

  _       DEMO      ADMIN                              Revoked

  _       DEMO      DEMO                               Granted

  _       DEMO      RESTRICT                           Granted

  _       DEMO      CST                                Granted

Domain: _____ * Group: _____ *

Direct command...: _____
(PF4=Add)      Clear    Disable    Enable    Grant    Purge     Revoke
Service   (PF5=flip)
```

The security status for the domain/group combination is displayed in the second header. The security status applied to the combination is not shown for each of the domain's services in `Current Access`. However, a status defined for the domain/group combination applies to associated services by default. Only statuses explicitly defined for individual services are displayed in `Current Access`.

3   Enter the appropriate code in `Act` for the service.

Use the Clear action to remove security settings for a service. This action also clears the service's settings from the security cache.

Method Level

Use the third level in the security hierarchy to allow the group access to specific methods of a service or to override settings defined at the service or domain/group level.

▶ **To set security options at the method level:**

1   Enter "S" in `Act` for the service in the Maintain Service Security by Domain window.

The Security for Domain window is displayed. For example:

```
BSOSO_MN  ***** Business Service Administration Subsystem *****
Mar  1,06        Security for Domain ADMIN/Group ADMIN: Revoked   11:29 PM


                                    Current  Previous
Act             Service             Access   Access
--- ------------------------------ -------- --------

  _   BusinessServicesAuditTrail
  _   BusinessServicesMethodAuditTrail
  _   Deployment
  _   Domain
  _   DomainAuditTrail
  _   Group
  _   GroupLevelSecurity
Object: _____
Direct command...: _____
Clear      Disable    Enable     Grant      Methods     Revoke
(PF5=flip)
```

The security status for the service is displayed in the second header (Revoked in the example above). The security status applied to the service is not shown for each of its methods in `Current Access`, since a status defined for a service applies to associated methods by default. Only a status defined for an individual method is displayed in `Current Access`.

2   Enter the appropriate code in `Act` for the method.

Use the Clear action to remove security settings for a method. This action also clears the method's settings from the security cache.

Security Statuses

How security statuses are interpreted when a client request is made to the dispatch server depends on the security exit you are using.

> **Note:** For more information, see **Use the Security Exit Modules**.

This section covers the following topics:

- No Privilege Status
- Revoked and Granted Status
- Disabled and Enabled Status

No Privilege Status

When a domain/group combination has no security option set, no privileges have been assigned to the combination. This is the status in effect after you add a new domain/group combination. You can set privileges for the domain's services and methods at this point. Or you can set a security option for the combination that will apply to all services and methods in the domain.

This scenario has a different meaning when it belongs to a service or method. When you set a domain/group combination's security status, all of its services and their methods "inherit" that status. However, their current security setting remains blank unless you assign statuses to them individually. When you explicitly set a service's security status, that status applies to its methods as well.

Revoked and Granted Status

The granted status allows the group access to the domain, service, or method to which the status is assigned. The revoked status prevents users from accessing the domain/group combination, service, or method to which the status is assigned.

There are different ways to interpret the granted and revoked settings, depending on the security exit you are using. For example, the default security exit (BS_USEC2) checks access privileges from the lowest level up (from method to domain/group combination) for each group to which a user belongs. A user can access a method that has the granted status, although the service to which the method belongs has the revoked status.

On the other hand, if you are using the BS_USEC1 security exit, the revoked status always takes precedence over the granted status. If the domain/group's status is granted, and the service's status is revoked, the service and its methods are inaccessible. Even if one of the methods of the service has access explicitly granted, the revoked status at the service level disallows access to any of the methods.

Disabled and Enabled Status

Disabling privileges at any level does not change the current security settings. Rather, it is a temporary override, usually for maintenance purposes. When you apply the disabled status to a domain/group combination, service, or method, the `Previous Access` field is populated with the value that was in the `Current Access` field. Once the override is no longer necessary, use the enable action to return the setting to the value in place when the disable status was applied. At any time, you can change the disable status to grant, revoke, or no privileges.

Use the Security Exit Modules

Natural Business Services provides exit modules (BS_USEC*) you can exchange or modify to interpret the security settings differently. These exits receive information from the Natural Business Services security server when a request is made to the server. The information includes the name of the user, the request being made, the user's groups, and the security settings for the appropriate domain. BS_USEC2 is the default security exit.

The security exit interprets the security settings and returns a flag indicating whether or not to grant access to the user for the request. If the request is not granted, a message number and three substitution values may be returned. (The text associated with the message number is defined in the SYSERR library.) The message data is returned to the client, written to the security log, and placed in the security cache for quicker access on subsequent calls.

The information supplied to the exit is defined in a PDA member called BS_USECA. This member is supplied in source format to provide context and additional documentation. It should not be changed or recompiled.

Tip: To switch security exits, access the Maintain Control Record window and replace the current exit. Then restart all security servers. For more information, see **Using the Business Service Control Record**.

This section covers the following topics:

- BS_USEC0 Module (Template)
- BS_USEC1 Module ("Pessimistic" Checking)

- BS_USEC2 Module ("Optimistic" Checking)

BS_USEC0 Module (Template)

This module is a template containing a data section that defines the required parameter data for the security exit. You can insert all the code required to implement security at the business service level into this template.

BS_USEC1 Module ("Pessimistic" Checking)

This module implements "pessimistic" security checking. It disallows the user access if any restrictions are found. The algorithm is:

```
If no privileges are defined at any level (domain, service, or method) then
  Set the access flag to 'No Privileges'
  Exit the routine
End If
If privileges are disabled at any level then
  Set the access flag to 'Disabled'
  Exit the routine
End If
If privileges are revoked at any level then
  Set the access flag to 'Revoked'
  Exit the routine
End If
If privileges are granted at any level then
  Set the access flag to 'Granted'
  Exit the routine
End If
Evaluate the access flag setting and set the SEC_GRANTED flag accordingly
```

BS_USEC2 Module ("Optimistic" Checking)

This module is the default security exit. It implements "optimistic" security checking. It allows the user access if any grant status is found for any group at the lowest level in the security hierarchy. The algorithm is:

```
If no privileges are defined (all matrix entries are 0) then
  Set the access flag to 'No Privileges'
  Exit the routine
Else
  For each group do
    If any level (domain, service, or method) is disabled then
      Set the access flag to 'Disabled'
      Go to the next group
    Else
      For each level from 3 to 1 (method to domain) do
        If the current level is revoked
          Set the access flag to 'Revoked'
          Go to the next group
```

```
      Else
        Set the access flag to 'Granted'
        Exit the routine
      End If
    End For
  End If
End For
End If

Evaluate the access flag setting and set the SEC_GRANTED flag accordingly
```

Use the Cache Synchronization Exit Module

Whenever security data is updated in the Administration subsystem, the affected data in the security cache is cleared. The next request from the client causes the security cache to be populated with up-to-date information.

Whenever service- or method-level security is updated, the cache synchronization exit module (BS_SYNCN) is called to clear the data from the cache. You can modify this module to override the level at which cache cleanup is performed. The source for the exit contains extensive comments to help you modify the routine to your requirements.

> **Note:** You cannot override levels if a change is made to security for a domain/group combination. All entries in the cache for the domain, regardless of group, are deleted.

By default, if a change is made to security at the service level, all entries for the domain/service combination are deleted from the cache. You can modify the exit to specify that all entries for the domain are to be deleted, regardless of service.

By default, if a change is made to security at the method level, all entries for the domain/service/method combination are deleted from the cache. You can modify the exit to specify that all entries for the domain or domain/service are deleted.

> **Tip:** Do not specify that deletion begin at a lower level than that being updated. For example, if a service's security is updated, do not specify deletion to occur at the method level. If you do so, or if no synchronization level is specified, the system defaults to the security level being updated.

> **Note:** For information on security cache synchronization, see **Synchronize the Security Cache and Cleanup Queue**.

9 Using Natural Business Services Tools

Natural Business Services supplies several tools on the client and on the server to help administer and maintain your business services environment. This section covers the following topics:

Client Tools

You can use several tools on the client to administer and maintain your business services environment.

- Broker Driver Program
- Server Monitor

Broker Driver Program

The Broker Driver program communicates with Broker and allows you to send and receive messages.

▶ **To access the Broker Driver program:**

1   Select Broker Driver from the Start menu.

The Broker Driver panel is displayed. For example:
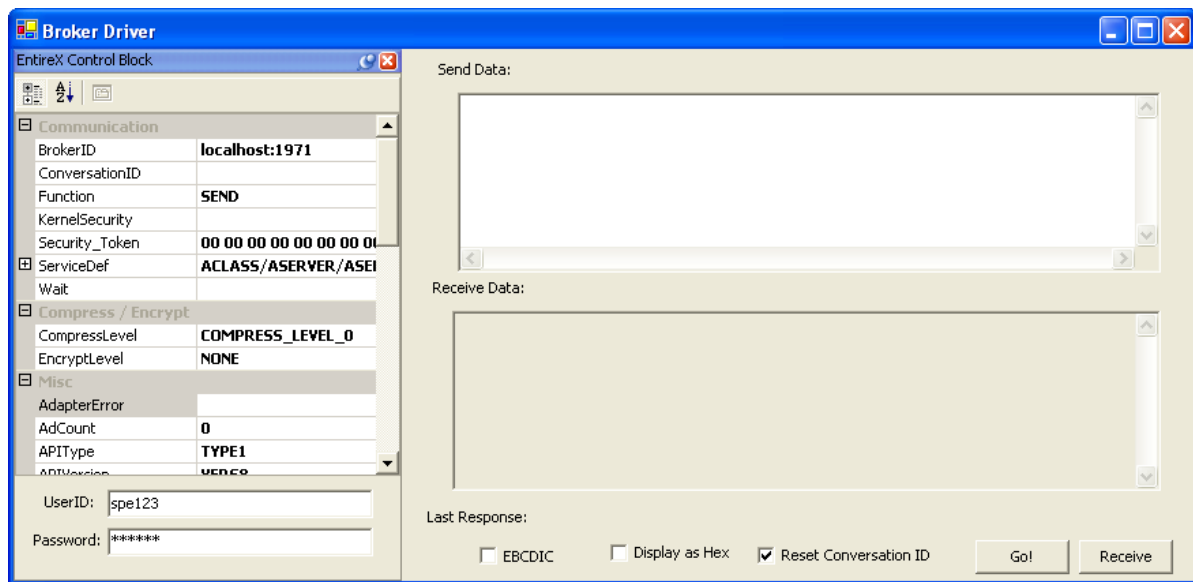
This panel displays information for the current connection.

2   Enter your user ID and password.

3   Select the data you want to send or receive.

4   Do one of the following:

- To send data to Broker, select **Go!**

- To receive data from Broker, select **Receive**.

Server Monitor

You can use this GUI application to manage servers. For information on using the Server Monitor, see NBS Server Monitor.

Server Tools

You can use several functions on the System Administration Functions menu to maintain the system and system data.

▶ **To access the System Administration Functions menu:**

1   Enter "SA" in `Function` on the Business Service Administration Subsystem main menu.

The System Administration main menu is displayed.

2   Enter "SF" in `Function`.

The System Administration Functions menu is displayed. For example:

```
BSSAMENU    ***** Business Service Administration Subsystem *****      CDLAYMN1
Feb 14               - System Administration Functions Menu -          09:32 PM


                  Functions

                  -------------------------------------------------


                  SQ    Synchronize Security Cache & Cleanup Queue



                  SC    Reset Security Cache



                  DC    Delete Communication Logs


```

```
                     ?    Help

                     .    Terminate

                     ------------------------------------------------

Function .......... __




Command ..........  _____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      help  retrn quit        flip                                      main
```

The following sections describe these functions and how to use them:

- Synchronize the Security Cache and Cleanup Queue
- Reset the Security Cache
- Delete the Communication Logs

Synchronize the Security Cache and Cleanup Queue

Use this function to complete any cache cleanup tasks that failed prior to completion, such as when clients performing remote CALLNATs receive an NBS5065> Request cannot be handled while cache cleanup is in progress error repeatedly. Since the cleanup record is only deleted after the operation completes successfully, you can run this function at any time and as often as necessary.

Security Cache

When a business service dispatch server receives a request from the client, the security server assigned to it checks the user's security privileges. If lookups had to be made to a user table, then a group table, then a business service definition, and then a security table for every request, the time required to process a request would be inordinately long. To optimize this process, the security server builds up a cache of security data when it first checks security for a given request. Only one table lookup is required to check the access rights of a user, which greatly improves response time after the initial request.

When data that affects the contents of the security cache is updated in the Administration subsystem, the affected data is cleared from the cache. For example, if a subprogram proxy is regenerated

using Natural Construct, or if an administrator revokes access privileges for a group, the appropriate data in the security cache is cleared. The next request from the client populates the security cache with up-to-date information.

Cleanup Queue

If an interruption occurs while the security cache is being updated, the update can fail. If this happens, the security cache will not be synchronized with the Administration subsystem files. The cleanup queue ensures this does not happen.

Before an update is made to the security cache, Natural Business Services writes a record to the cleanup queue containing information about the impending update. If the update fails, the record remains in the queue so you can synchronize the security cache with the Administration subsystem files. If the update completes successfully, the record in the cleanup queue is automatically deleted.

Reset the Security Cache

Use this function to completely clear all entries from the security cache. By emptying the cache, the security server is forced to rebuild all security cache entries.

After specifying a request to clear the cache, you may receive an additional choice. If the cache contains more than 1000 records, choose one of the following:

- Submit the cache delete as a batch job (recommended, especially for large caches)
- Run the cache delete online (this may take a long time if the cache is large)

> **Tip:** Use the Reset Security Cache function if you change from using the Administration subsystem security files (User, Group) to using Natural Security data directly, or vice versa. It clears the cache and allows the cache to be rebuilt based on the new user and group definitions.

> **Note:** If you change the security mode in a dispatch server or change the SAF resource profiles, reset the security cache to clear the old settings.

Delete the Communication Logs

Use this function to delete communication log file records by date (both Broker and Natural Development Server log file records). If there are only a few records, you can run this utility online. However, it is recommended that you run the utility in batch mode as it is easier to keep track of which records were deleted.

▶ **To clean the communication logs in batch mode:**

1   Execute the BS_BKLOG subprogram.

The Delete Communication Log Records by Date panel is displayed. For example:

```
BS_BKLOG        Business Service Administration Subsystem           BS_BKLG1
Feb 28,06              Delete Communication Log Records by Date
09:38 M




              Please Enter Start Date: _____  (YYYYMMDD)

                              End   Date: _____  (YYYYMMDD)



              Do you need a hard copy : _



              Do you need a WORK FILE : _
```

> **Note:** This panel is also displayed when you run the utility in online.

2   Enter the start and end dates in *YYYYMMDD* format.

3   Optionally, mark the appropriate field to print a hardcopy report or copy a report to a work file.

10 Domains Supplied with Natural Business Services

Natural Business Services uses several business services supplied in the INTERNAL and ADMIN domains. You can also take advantage of these services, as well as those in the DEMO domain.

This section covers the following topics:

ADMIN Domain

Natural Business Services supplies a group of business services that are only required by an administrator. These business services allow the consumer to grant security to the appropriate groups of users and perform other tasks, including:

- Check audit trails
- Deploy services
- Modify security information

This section describes the business services contained in the ADMIN domain:

- BusinessServicesAuditTrail (Version 010101)
- BusinessServicesMethodAuditTrail (Version 010101)
- Domain (Version 010101)
- DomainAuditTrail (Version 010101)
- Deployment (Version 010101)
- Group (Version 010101)
- GroupLevelSecurity (Version 010101)
- MethodLevelSecurity (Version 010101)
- SecurityExceptionsReport (Version 010101)
- SecurityPermissionsReport (Version 010101)
- StepLibraryAuditTrail (Version 010101)
- ServiceLevelSecurity (Version 010101)

BusinessServicesAuditTrail (Version 010101)

This service retrieves a record of changes to business services in descending order. It has the following methods:

Method Name	Description
FindByDateTimeDomainDesc	Retrieves a record of changes to business services in descending order by date, time, and domain.
FindByDomainServiceVrsDateDesc	Retrieves a record of changes to business services in descending order by domain, service, version, date, and time.

BusinessServicesMethodAuditTrail (Version 010101)

This service retrieves a record of changes to service methods in descending order. It has the following methods:

Method Name	Description
FindByDateTimeDomainDesc	Retrieves a record of changes to service methods in descending order by date, time, and domain.
FindByDomainServiceVrsDateDesc	Retrieves a record of changes to service methods in descending order by domain, service, version, date, and time.

Domain (Version 010101)

This service retrieves and maintains a set of domains. It has the following methods:

Method Name	Description
Delete	Deletes row(s) as part of the MultiMaint method.
FindByDomainName	Retrieves row(s) by domain.
FindBySteplibName	Retrieves row(s) by step library chain.
MultiMaint	Performs methods on multiple domains based on the state of the rows.
Store	Adds row(s) as part of the MultiMaint method.
Update	Updates row(s) as part of the MultiMaint method.

DomainAuditTrail (Version 010101)

This service retrieves a record of changes to domains in descending order. It has the following methods:

Method Name	Description
FindByDateTimeDomainDesc	Retrieves a record of changes to domains in descending order by date, time, and domain.
FindByDomainDateTimeDesc	Retrieves a record of changes to domains in descending order by domain, date, and time.

Deployment (Version 010101)

This service deploys domains, associated services, and step library chains between environments. It contains the Services method.

You must specify the DBID (database identification) and FNR (file number) for the source and target environments, as well as a range value for domain records to deploy. You may use wildcards to indicate the range (for example, "*", ">", and "<").

Group (Version 010101)

This service retrieves and maintains a set of groups. It has the following methods:

Method Name	Description
Delete	Deletes row(s) as part of the MultiMaint method.
FindByGroup	Retrieves row(s) by group.
MultiMaint	Performs methods on multiple groups based on the state of the rows.
Store	Adds row(s) as part of the MultiMaint method.
Update	Updates row(s) as part of the MultiMaint method.

GroupLevelSecurity (Version 010101)

This service searches the domain/group combinations and allows you to set security permissions (see the Security method for valid states). The GroupLevelSecurity service has the following methods:

Method Name	Description
AddDomainGroupCombination	Adds a domain/group combination. Specify the domain/group names in the ASOSGK input parameter group.
FindDomainGroup	Retrieves security information for groups by domain and group name.
FindGroupDomain	Retrieves security information for groups by group and domain name.
Security	Adjusts security permissions on the rows that provide valid row states. Valid row states are: ■ CL (clear) ■ DA (disable) ■ EN (enable) ■ GR (grant) ■ RE (revoke)

Method Name	Description
DeleteDomainGroupCombination	Deletes a domain/group combination. Specify the domain/group name in the ASOSGK input parameter group.
FindGroupsWithinDomain	Retrieves group-level security information for a domain. Specify the domain name in the ASOSGK input parameter group.
FindSpecificDomainGroup	Retrieves group-level security information for a domain/group combination. Specify the domain/group name in the ASOSGK input parameter group.

MethodLevelSecurity (Version 010101)

This service searches the business services within a domain and allows you to set security permissions (see the Security method for valid states). Security is set based on the group ID. The Method-LevelSecurity service has the following methods:

Method Name	Description
FindByDomainMethodServiceVersion	Retrieves security information for methods by domain, method, service, and version.
FindByDomainServiceMethod	Retrieves security information for methods by domain, service, and method.
FindByDomainServiceVersionMethod	Retrieves security information for methods by domain, service, version, and method.
FindByMethodServiceVersion	Retrieves security information for methods by method, service, and version.
FindMethodsWithinDomainService	Retrieves service-level security information for a domain/service combination. Specify the domain/service name in the DOM2BKEY input parameter group.
Security	Adjusts security permissions on the rows that provide valid row states. Valid row states are: ■ CL (clear) ■ DA (disable) ■ EN (enable) ■ GR (grant) ■ RE (revoke)

SecurityExceptionsReport (Version 010101)

This service retrieves runtime security errors in descending order by date and time. It contains the FindByDateTimeDesc method.

SecurityPermissionsReport (Version 010101)

This service retrieves the business services security report. It has the following methods:

Method Name	Description
FindByDomain	Retrieves the business services security report by domain.
FindByDomainGroupServiceMethod	Retrieves the business services security report by domain, group, service, and method.
FindByGroup	Retrieves the business services security report by group.
FindByGroupDomainServiceMethod	Retrieves the business services security report by group, domain, service, and method.

StepLibraryAuditTrail (Version 010101)

This service retrieves a record of changes to the step library chains in descending order. It has the following methods:

Method Name	Description
FindByDateTimeStepLibNameDesc	Retrieves a record of changes to the step library chains in descending order by date, time, and step library chain.
FindByStepLibNameDateTimeDesc	Retrieves a record of changes to the step library chains in descending order by step library chain, date, and time.

ServiceLevelSecurity (Version 010101)

This service searches the methods within a business service and allows you to set security permissions (see the Security method for valid states). Security is set based on the group ID. The ServiceLevelSecurity service has the following methods:

Method Name	Description
FindByDefaultProxy	Retrieves security information for business services by default subprogram proxy.
FindByDomain	Retrieves security information for business services by domain.
FindByDomainService	Retrieves security information for business services by domain and service.
FindBySteplibName	Retrieves security information for business services by step library chain.

Method Name	Description
Security	Adjusts security permissions on the rows that provide valid row states. Valid row states are: ■ CL (clear) ■ DA (disable) ■ EN (enable) ■ GR (grant) ■ RE (revoke)
ServiceCountByDomain	Counts business service rows by domain.
FindServicesWithinDomain	Retrieves service-level security information for a domain. Specify the domain name in the DOM2BKEY input parameter group.

DEMO Domain

The DEMO domain contains the business services used in the demo application, SYSBIZDE. This section describes the business services supplied in the DEMO domain:

- Calculator (Version 010101)
- CalculatorAdvance (Version 010101)
- Customer (Version 010101)
- CustomerCreditAnalysis (Version 010101)
- CustomerWithContactData (Version 010101)
- CustomerWithContactData (Version 020101)
- ErrorMessageTesting (Version 010101)
- FlipString (Version 010101)
- GreatestCommonDenominator (Version 010101)
- Order (Version 010101)
- Order (Version 020101)
- Product (Version 010101)
- StringManipulation (Version 010101)

- Warehouse (Version 010101)

Calculator (Version 010101)

This service performs simple calculator functions. In addition, a success flag is set if the solution is greater than the success criteria. The Calculator service has the following methods:

Method Name	Description
Add	Adds two numbers together.
Divide	Divides two numbers.
Multiply	Multiplies two numbers.
Subtract	Subtracts one number from another.

CalculatorAdvance (Version 010101)

This service provides a calculator that performs advanced calculations. It was generated using the Object-Generic-Subp model and has the following methods:

Method Name	Description
Add	Adds two numbers together.
Divide	Divides two numbers.
Multiply	Multiplies two numbers.
Subtract	Subtracts one number from another.
GreatestCommonDenominator	Executes the GCDN subprogram to calculate the greatest common divisor for two numbers.
SolutionWithLowerNumbers	Executes the CALC and GCDN subprograms to calculate the solution with the lowest numbers.

Customer (Version 010101)

This service retrieves customers and maintains a customer. It has the following methods:

Method Name	Description
BROWSE	Lists customers in the Customer file.
DELETE	Deletes a customer record.
EXISTS	Determines whether a customer exists.
FORMER	Retrieves the previous customer record in the file.
GET	Retrieves a customer record.
INITIALIZE	Initializes a customer record.

Method Name	Description
NEXT	Retrieves the next customer record in the file.
STORE	Adds a customer record to the file.
UPDATE	Updates a customer record.

■ To maintain contact information for one customer, use the CustomerWithContactData service, version 010101.

■ To maintain contact information for a set of customers, use the CustomerWithContactData service, version 020101.

■ To flag customers with credit limits greater than $10,000, use the CustomerCreditAnalysis service, version 010101.

CustomerCreditAnalysis (Version 010101)

This service retrieves customers and maintains a customer. It also flags customers with credit limits greater than $10,000.

CustomerWithContactData (Version 010101)

This service retrieves customers and maintains a customer and contact information, such as phone numbers and email addresses.

CustomerWithContactData (Version 020101)

This service retrieves and maintains a set of customers and their contact information, such as phone numbers and email addresses. It has the following methods:

Method Name	Description
Delete	Deletes row(s) as part of the MultiMaint method.
FindByBusinessName	Retrieves row(s) by business name.
MultiMaint	Performs individual methods on multiple rows based on the state of the rows.
Store	Adds row(s) as part of the MultiMaint method.
Update	Updates row(s) as part of the MultiMaint method.
FindByCustomerNumber	Retrieves row(s) by customer number.
FindByCustomerWarehouseId	Retrieves row(s) by customer and warehouse ID.

ErrorMessageTesting (Version 010101)

This service tests Natural error messages. To trigger an error message, type `NATnnnn` (for example, NAT3009) in the input field.

FlipString (Version 010101)

This service retrieves a text string and reverses the order of characters in the string (for example, ABC becomes CBA).

GreatestCommonDenominator (Version 010101)

This service calculates the greatest common divisor for two numbers.

Order (Version 010101)

This service retrieves orders and maintains an order. It has the following methods:

Method Name	Description
BROWSE	Lists orders in the Order file.
DELETE	Deletes an order record.
EXISTS	Determines whether an order exists.
FORMER	Retrieves the previous order record in the file.
GET	Retrieves an order record.
INITIALIZE	Initializes an order record.
NEXT	Retrieves the next order record in the file.
STORE	Adds an order record to the file.
UPDATE	Updates an order record.

Order (Version 020101)

This service retrieves and maintains a set of orders. It has the following methods:

Method Name	Description
FindByCustomerDate	Retrieves row(s) by customer number and date.
FindByOrderCustomerNumber	Retrieves row(s) by order and customer number.
FindByOrderNumber	Retrieves row(s) by order number.
FindByOrderWarehouseId	Retrieves row(s) by order number and warehouse ID.
OrderWarehouseIdCount	Counts order row(s) based on order number and warehouse ID.

Product (Version 010101)

This service retrieves products and maintains a product. It has the following methods:

Method Name	Description
BROWSE	Lists products in the Product file.
DELETE	Deletes a product record.
EXISTS	Determines whether a product exists.
FORMER	Retrieves the previous product record in the file.
GET	Retrieves a product record.
INITIALIZE	Initializes a product record.
NEXT	Retrieves the next product record in the file.
STORE	Adds a product record to the file.
UPDATE	Updates a product record.

StringManipulation (Version 010101)

This service manipulates the characters in a text string. It has the following methods:

Method Name	Description
ConvertToLowerCase	Executes the CSUCASE subprogram to convert a text string to lower case.
ConvertToMixedCase	Executes the CSUCASE subprogram to convert a text string to mixed case.
ConvertToUpperCase	Executes the CSUCASE subprogram to convert a text string to upper case.
ReverseString	Executes the FLIPSTR subprogram to reverse the order of characters in a text string (for example, ABC becomes CBA).

Warehouse (Version 010101)

This service retrieves warehouses and maintains a warehouse. It has the following methods:

Method Name	Description
BROWSE	Lists warehouses in the Warehouse file.
DELETE	Deletes a warehouse record.
EXISTS	Determines whether a warehouse exists.
FORMER	Retrieves the previous warehouse record in the file.
GET	Retrieves a warehouse record.
INITIALIZE	Initializes a warehouse record.
NEXT	Retrieves the next warehouse record in the file.

Method Name	Description
STORE	Adds a warehouse record to the file.
UPDATE	Updates a warehouse record.

INTERNAL Domain

For a developer to use functionality within Natural Business Services, he or she must belong to a group that has access to the INTERNAL domain. The business services in the INTERNAL domain perform many functions, including:

- Access the Business Service repository to determine what services are available.
- Create XML that describes the business service.
- Read Natural source code.

This section describes the business services contained in the INTERNAL domain:

- BusinessService (Version 010101)
- BusinessServiceDetail (Version 010101)
- BusinessServiceDetail (Version 020101)
- GenerateServiceDataDefinition (Version 010101)
- ModelDefaults (Version 010101)
- NaturalLibrary (Version 010101)
- NaturalModule (Version 010101)
- NaturalSource (Version 010101)
- NBSlfiles (Version 010101)
- PredictMetaData (Version 010101)
- RepositoryQueries (Version 010101)
- ServiceProxyMetaData (Version 010101)
- StepLibraryChain (Version 010101)
- SubprogramPDAs (Version 010101)

BusinessService (Version 010101)

This service retrieves a set of business services. It has the following methods:

Method Name	Description
ServiceCountByDomain	Counts row(s) by domain.
FindByDefaultProxy	Retrieves row(s) by default subprogram proxy.
FindByDomain	Retrieves row(s) by domain.
FindByDomainService	Retrieves row(s) by domain and service.
FindBySteplibName	Retrieves row(s) by step library chain.

BusinessServiceDetail (Version 010101)

This service retrieves details about a business service, such as the methods and step library chain used. It contains the Retrieve method.

BusinessServiceDetail (Version 020101)

This service retrieves and maintains details about a business service, such as the methods and step library chain used. It has the following methods:

Method Name	Description
Add	Executes the BSIF___N subprogram with the Add method.
Delete	Executes the BSIF___N subprogram with the Delete method.
Get	Executes the BSIF___N subprogram with the Get method.
Next	Executes the BSIF___N subprogram with the Next method.
Update	Executes the BSIF___N subprogram with the Update method.
Retrieve	Executes the SNUSINFO subprogram with the Retrieve method.
RetrieveProxyAndSubprogram	Accepts a domain name, service name, and version number and returns all subprogram proxies and their associated subprograms.

GenerateServiceDataDefinition (Version 010101)

This service accepts the name of a subprogram proxy and generates the service parameter definitions. It contains the Generate method.

ModelDefaults (Version 010101)

This service retrieves the default values for a model. It has the following methods:

Method Name	Description
ObjectBrowseSelectMethods	Accepts the name of an object browse subprogram and returns user-modifiable method names for the service.
ObjectGenericLevel1s	Accepts the names of up to 10 subprograms and returns a maximum of 100 level 1 fields for each.

NaturalLibrary (Version 010101)

This service lists all Natural libraries based on the database ID (DBID) and file number (FNR). It contains the List method.

NaturalModule (Version 010101)

This service retrieves a list of Natural modules or determines if a module exists. It has the following methods:

Method Name	Description
Browse	Lists Natural modules. You can restrict the list to proxies that are not subprograms (OPT-TYPE=Y) or to subprograms that are not proxies (OPT-TYPE=X).
Exist	Determines whether a Natural module exists.

NaturalSource (Version 010101)

This service downloads or uploads Natural source code. It has the following methods:

Method Name	Description
Download	Downloads Natural source code.
Upload	Uploads Natural source code.

NBSIfiles (Version 010101)

This service retrieves the DBID (database identification) and FNR (file number) for LFILE 135 and 136. It contains the Retrieve method.

PredictMetaData (Version 010101)

This service retrieves field and relationship information for files in Predict. It has the following methods:

Method Name	Description
Download	Downloads field and relationship information for a file defined in Predict.
RetrievePredictFields	Retrieves Predict metadata for all fields in a file.
RetrievePredictFiles	Retrieves a list of Predict files.

RepositoryQueries (Version 010101)

This service retrieves information from the Business Service repository, such as which business services use the Browse method. It has the following methods:

Method Name	Description
FindAllUpperCaseBrowses	Retrieves business services that use the BROWSE method.
FindBrowses	Retrieves business services that use the Browse method.
FindByDomainMethodServiceVersion	Accepts the name of a domain and retrieves services by domain, method, service, and version.
FindByDomainServiceMethod	Retrieves services by domain, service, and method. **Note:** This method not required by current client queries.
FindByDomainServiceVersionMethod	Retrieves services by domain, service, version, and method. **Note:** This method not required by current client queries.
FindByMethodServiceVersion	Retrieves services by method, service, and version. **Note:** If you specify the name of a domain, this method will retrieve services in domain order.
FindByServiceVersion	Retrieves services by service and version. **Note:** If you specify the name of a domain, this method will retrieve services in domain order.
FindFindBys	Retrieves services that have method names beginning with "FindBy".
FindMultiMaints	Retrieves services that have a method called "MultiMaint".

ServiceProxyMetaData (Version 010101)

This service uses the conversation factory to download the metadata for a subprogram proxy. It contains the Download method.

StepLibraryChain (Version 010101)

This service retrieves and maintains a set of step library chains. It has the following methods:

Method Name	Description
Delete	Deletes row(s) as part of the MultiMaint method.
FindByStepLibraryChain	Retrieves row(s) by step library chain.
MultiMaint	Performs individual methods on multiple rows based on the state of the rows.
Store	Adds row(s) as part of the MultiMaint method.
Update	Updates row(s) as part of the MultiMaint method.

SubprogramPDAs (Version 010101)

This service accepts a list of parameter data areas (PDAs) and the date on which they were last compiled and downloads all parameters that have been modified since that date. It contains the Download method.

11 Auditing Business Services

The Business Service Administration subsystem provides several queries and logs you can use.

💡 **Tip:** You can also use the **Service Audit** window on the client to audit business services. For information on this window in the Eclipse plug-in, see Audit Business Services. For information on this window in the Natural plug-in, see Audit Business Services.

This section covers the following topics:

Queries and Logs for Servers

You can view query and log query panels containing information about servers used with Natural Business Services. This section covers the following topics:

- Access the Query Panels
- Access the Log Query Panels

Access the Query Panels

▶ **To access the query panels:**

1 Enter "SA" in Function on the Business Service Administration Subsystem main menu.

The System Administration main menu is displayed.

2 Enter "QM" in Function.

The System Administration Query menu is displayed. For example:

```
BSSQMENS   ***** Business Service Administration Subsystem *****     CDLAYMN1
Feb 14,06              - System Administration Query Menu -          09:35 PM


                   Functions

                   ------------------------------------------------

                   SE   Query Services



                   CL   Query Communication Logs
```

```
                       GR   Query Group Table

                       US   Query User Table

                       UG   Query User's Groups



                       ?    Help

                       .    Terminate

                       ----------------------------------------------

Function .......... __




Command ........... _____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      help  retrn quit         flip                                     main
```

3 Enter a function code in Function to display the corresponding query or log panel.

The function codes and functions are:

Function Code	Function	Description
SE	Query Servers	Displays existing servers and their characteristics as defined on the Maintain Servers panels.
CL	Query Communications Logs	Displays messages that occurred during the operation of servers.

Access the Log Query Panels

▶ **To access the log query panels:**

1 Enter "SA" in Function on the Business Service Administration Subsystem main menu.

The System Administration main menu is displayed.

2 Enter "LM" in Function.

The System Administration Log Query menu is displayed. For example:

```
BSSLMENU   ***** Business Service Administration Subsystem *****      CDLAYMN1
Feb 14,06             - System Administration Log Query Menu -          09:36 PM


                   Functions


                   ------------------------------------------------

                   LT    Log by Time Query Menu

                   LV    Log by Value Query Menu

















                   ?    Help

                   .    Terminate

                   ------------------------------------------------

Function .......... __




Command .......... _____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      help  retrn quit       flip                                      main
```

3   Enter a function code in Function to display the corresponding query or log query menu.

The function codes and functions are:

Function Code	Function	Description
LT	Log by Time Query menu	Invokes the Log by Time Query menu, where you can view Server logs and Control Record logs. The log information is presented in time order, beginning with the most recent log.
LV	Log by Value Query menu	Invokes the Log by Value Query menu, where you can view server logs. The values (for example, business service names) are presented in alphanumeric order, along with log information such as system activities and changes to records, messages, dates, and times.

Queries and Logs for User and Group Tables

You can view queries and logs containing information about the user and group tables. This section covers the following topics:

- Under Natural Business Services Security
- Under Natural Security

Under Natural Business Services Security

When only using Natural Business Services security, you can view queries for the group and user tables and logs of the activities performed on the tables.

Access the Query Panels

▶ **To access the user or group table query panels:**

1 Enter "SA" in `Function` on the Business Service Administration Subsystem main menu.

 The System Administration main menu is displayed.

2 Enter "QM" in `Function`.

 The System Administration Query menu is displayed (for an example of this menu, see **System Administration Query Menu**).

3 Enter a function code in `Function` to display the corresponding query or log panel.

The function codes and functions are:

Function Code	Function	Description
GR	Query Group Table	Displays the List Groups window, showing existing groups and their characteristics. Enter a group ID or starting value in the appropriate field to scroll to particular groups.
US	Query User Table	Displays the List Users window, showing existing users of type "P" and their characteristics. Enter a user ID or starting value in the appropriate field to scroll to a particular user.

Access the Log Query Panels

▶ **To access the log query panels:**

1 Enter "SA" in `Function` on the Business Service Administration Subsystem main menu.

The System Administration main menu is displayed.

2 Enter "LM" in `Function`.

The System Administration Log Query menu is displayed (for an example of this menu, see **System Administration Log Query Menu**).

3 Enter a function code in `Function` to display the corresponding query or log query menu.

The function codes and functions are:

Function Code	Function	Description
LT	Log by Time Query menu	Invokes the Log by Time Query menu, where you can view log information, such as actions, dates, and times. This information is presented in time order, beginning with the most recent log.
LV	Log by Value Query menu	Invokes the Log by Value Query menu, where you can view server logs. The values (for example, group names) are presented in alphanumeric order, along with log information, such as actions, dates, and times.

Under Natural Security

When using Natural Security, you can view query panels for the group and user tables and the user groups.

Access the Query Panels

▶ **To access the query panels under Natural Security:**

1 Enter "SA" in `Function` on the Business Service Administration Subsystem main menu.

   The System Administration main menu is displayed.

2 Enter "QM" in `Function`.

   The System Administration Query menu is displayed (for an example of this menu, see **System Administration Query Menu**).

3 Enter a function code in `Function` to display the corresponding query or log panel.

   The function codes and functions are:

Function Code	Function	Description
GR	Query Group Table	Displays the List Groups window, showing existing groups and their characteristics. Enter a group ID or starting value in the appropriate field to scroll to particular groups.
US	Query User Table	Displays the List Users window, showing existing users of type "P" and their characteristics. Enter a user ID or starting value in the appropriate field to scroll to a particular user.
UG	Query User's Groups	Displays the List Groups for a User window. ■ Enter a user ID in `User ID` to display the groups to which the user belongs. ■ Enter a starting value in `Start group value` to scroll to a particular group.

Queries and Logs for Natural Business Services Security

You can view queries and logs containing information about Natural Business Services security settings.

Access Query and Log Panels

▶ **To access the query and log panels for Natural Business Services security settings:**

1   Enter "AA" in `Function` on the Business Service Administration Subsystem main menu.

The Application Administration main menu is displayed.

2   Enter "QM" in `Function`.

The Application Administration Query menu is displayed. For example:

```
BSAQMENU   ***** Business Service Administration Subsystem *****      CDLAYMN1
Feb 15,06          - Application Administration Query Menu -          10:26 PM


                   Functions

                   ------------------------------------------------

                   AS   Query Application Service Definitions



                   OG   Query Object Security by Group

                   OD   Query Object Security by Domain

                   SC   Query Security Logs



                   DO   Query Domain Table

                   ST   Query Steplib Table



                   ?    Help

                   .    Terminate
```

```
                    -------------------------------------------------

  Function .......... __




  Command ........... _____
  Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
        help  retrn quit         flip                                      main
```

3   Enter the appropriate function code in `Function` to display the corresponding query panel.

The available functions are:

Function Code	Function	Description
OG	Query Service Security by Group	Displays groups in alphanumeric order, as well as their domains, services, methods, and security settings.
OD	Query Service Security by Domain	Displays domains in alphanumeric order, as well as the groups, services, methods, and security settings.
SC	Query Security Logs	Displays security messages generated in response to errors. Messages are listed in date and time order, beginning with the most recent message.

12 Deploying the Administration Subsystem

This section describes the steps to deploy the Business Service Administration subsystem in development and production environments. The following topics are covered:

Introduction

Deploying a business service is similar to distributing a traditional Natural application. However, the Natural Business Services runtime environment must be installed on each machine. This installation includes:

▪ All modules related to the Administration subsystem.

▪ All modules related to the Natural Business Services dispatch, security, and attach servers.

▪ One or more copies of the system files.

▪ At least one dispatch, security, and attach server.

Each runtime environment accesses the same centrally-located data by means of direct calls through one of the following:

▪ Entire Net-Work

▪ EntireX configured to use TCP/IP as the network transport protocol

▪ Natural Business Services security server (in the case of secure data)

Sample Deployment Scenario

The following diagram shows a scenario with one development and two production environments, one of which is on a remote machine. The Administration subsystem is installed on both machines, and Production environment 1 contains a copy of the Natural Business Services system files. In this scenario, Entire Net-Work is installed on both machines. EntireX is installed on the machine with both the Development environment and Production environment 1.
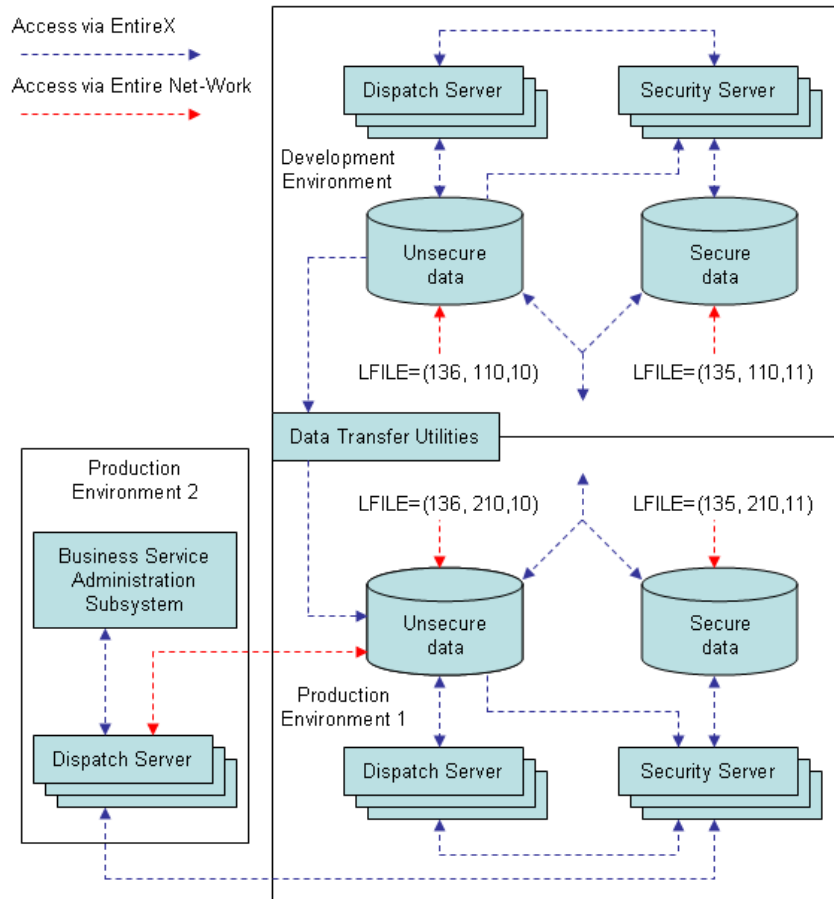
The sample scenario shows the option of having Natural Security installed on the machine that houses the Development environment and Production environment 1 to control users' access to secure data. To use Natural Security on the remote machine, another copy of Natural Security must be installed on that machine, a security server is required, and both copies of the Natural Security software must be configured to update the Natural Business Services system files with security information.

Create the Sample Deployment Scenario

This section describes the prerequisites and tasks involved in setting up the sample scenario shown in the previous section. Refer to the following steps when planning your own deployment strategy:

- Step 1: Install Software in Production Environments
- Step 2: Install Natural Business Services on Each Machine
- Step 3: Copy Natural Business Services System Files
- Step 4: Set up the System File Database in Entire Net-Work (Optional)
- Step 5: Define and Start an Attach Server
- Step 6: Define and Start Dispatch and Security Servers

Step 1: Install Software in Production Environments

Ensure that the following software is installed in the production environments:

- EntireX must be installed on one or more machine. In the sample scenario, it is installed on the machine containing both development and production environments.

- Either Entire Net-Work must be installed on every PC that will use the business service, or EntireX must be installed and configured to use TCP/IP as the network transport protocol.

- If you are not using EntireX configured for TCP/IP, Entire Net-Work must be installed on each server machine.

- Natural must be installed on each server machine.

- Adabas must be installed on each server machine.

> **Notes:**

1. For information on product versions, see Prerequisites.

2. For information on installing and configuring these products, see their respective installation guides.

Step 2: Install Natural Business Services on Each Machine

The Business Service Administration subsystem defines and manages Natural Business Services servers and maintains Administration subsystem data. You can access a full set of queries and log data in the subsystem.

Natural Business Services must be installed on each computer. The sample scenario in **Sample Deployment Scenario** shows the Administration subsystem installed on the machine that hosts the Development environment and Production environment 1 and on the remote computer that houses Production environment 2.

For information on installing Natural Business Services, see:

- Natural Business Services Installation on Mainframes

- Natural Business Services Installation on Unix

- Natural Business Services Installation on Windows

Step 3: Copy Natural Business Services System Files

In the example scenario, Production environment 1 shares the Administration subsystem with the Development environment. While it is possible to share the Natural Business Services system files between development and production environments, most organizations will want to create separate copies of these files to better ensure the integrity of their production data.

The files are accessed through logical file numbers, and they can be installed using any available physical file numbers. You can assign the logical file settings dynamically using the LFILE parameter, or you can link them into the Natural nucleus using the NTFILE parameter. Both files must be accessible from the Natural Business Services dispatch server, security server, and Administration subsystem.

Logical File 135

Logical file 135 contains all information concerning users, their security privileges, and the security cache. This information is stored separately from other dispatch server information to allow this sensitive data to be enciphered, if desired.

Logical File 136

Logical file 136 contains all information used by Natural Business Services, except the user and security information stored on logical file 135.

Step 4: Set up the System File Database in Entire Net-Work (Optional)

If the environment uses Entire Net-Work for its network transport protocol, ensure that the network administrator has set up the Natural Business Services system file database so that it is accessible to all appropriate clients.

Step 5: Define and Start an Attach Server

At runtime, an attach server launches other servers as they are needed. If possible, configure your system so that an attach server is started as part of the operating system activation process.

> **Tip:** The attach server can automatically initiate multiple dispatchers during startup. For information, see **Initiate Multiple Dispatchers During Startup**.

Start an Attach Server as a Started Task

Typically, you start attach servers in an online environment, which is resource intensive. If desired, you can start an attach server as a started task. Although you will be starting the server from the console, as opposed to from within Natural Business Services, you can continue to use all other features in the Business Service Administration system.

▶ **To run an attach server as a started task:**

1   Set up the appropriate JCL. For example:

```
//*******************************************************************
//*
//ADA148   EXEC PGM=ADA148,REGION=2000K,PARM='922'
//STEPLIB  DD   DISP=SHR,DSN=OPS.COMN.LOAD
//         DD   DISP=SHR,DSN=RZDBA.DB922.NEWLOAD
//         DD   DISP=SHR,DSN=RZDBA.DB922.LOAD
//SYSUDUMP DD   SYSOUT=X
//DDCARD DD *
ADARUN  PROGRAM=USER,DATABASE=13001,MODE=MULTI,SVC=249
//DDPRINT  DD   SYSOUT=*
//DDDRUCK  DD   SYSOUT=*
//*
//ATTCHPRD EXEC PGM=NATSPE31,REGION=7000K,
//         PARM='IM=D,PROFILE=NBS511'
//STEPLIB  DD DSN=PPEX.NATURAL.LOAD,DISP=SHR
//         DD DSN=RZDBA.DB922.NEWLOAD,DISP=SHR
//         DD DSN=RZDBA.DB922.LOAD,DISP=SHR
//SORTLIB   DD DUMMY
//SORTOUT   DD DUMMY,DCB=BLKSIZE=80
//DDSORTIN  DD DISP=(,PASS),DSN=&&SORT,UNIT=SYSDA,
//             DCB=RECFM=FB,SPACE=(CYL,(5,5))
//DDSORTUT  DD DISP=(OLD,DELETE),DSN=*.DDSORTIN,VOL=REF=*.DDSORTIN
//SORTWK01  DD UNIT=SYSDA,SPACE=(CYL,5)
//SORTWK02  DD UNIT=SYSDA,SPACE=(CYL,5)
//SORTWK03  DD UNIT=SYSDA,SPACE=(CYL,5)
//SORTWK04  DD UNIT=SYSDA,SPACE=(CYL,5)
//SORTMSG   DD SYSOUT=X
//CMPRINT   DD SYSOUT=X,DCB=BLKSIZE=132
//CMPRT01   DD SYSOUT=X
//CMWKF01   DD SYSOUT=X
//CMWKF07   DD DSN=&&TEMP,UNIT=SYSDA,SPACE=(CYL,(1,1)),DISP=(,DELETE)
//CMPRT02   DD SYSOUT=X
//CMPRT03   DD SYSOUT=X
//CMEDIT31  DD DSN=DAEFEDT.V31.CMEDIT,DISP=SHR
//CMSYNIN   DD DISP=SHR,DSN=NBS511.JCL(business service input data)
//DDCARD DD *
ADARUN  PROGRAM=USER,DATABASE=13001,MODE=MULTI,SVC=249
```

2   Initiate the attach server from the Business Service Administration main menu.

3   Enter "%%" to display the JCL submitted to the editor.

In the example, the JCL is contained in `business service input data`.

4   Scroll to the information following the LOGON. For example:

```
LOGON SYSBIZ
SPSSTART
ATTACH-QA,%
CNDSHE,%
%*
,%
O,%
O,%
  ,%
  ,%
  ,%
OOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOO,%




FIN
```

5   Use this information to supply the required information in the JCL for the started task.

Step 6: Define and Start Dispatch and Security Servers

At runtime, the dispatch server is responsible for invoking the Natural business services requested by the client. This server accesses the Natural Business Services system files for self-configuration on startup and for writing messages. It also uses the Natural Business Services security server (defined as one of its remote services) to determine the validity and location of the requested target service.

The dispatch server communicates with the security server to validate users and check their privileges and requests for business services. At least one Natural Business Services security server must be available to an environment. It can be installed on any machine that has access to the Business Service Administration subsystem data. However, to minimize network traffic, the Natural Business Services security server should run in the same location as the Administration subsystem data.

Normally, dispatch and security servers are started up automatically on demand by an attach server.

> **Note:** For information on defining servers, see **Defining and Managing Servers**.

Under Natural Security

Natural Business Services security servers can be configured to work with Natural Security so that you can use Natural Security users and groups instead of Natural Business Services tables. If you are using Natural Security, the dispatch and security servers must have access to Natural Security data. When Natural Security is operating, the dispatch server makes calls to Natural Security to help determine client authorizations, and the dispatch server "impersonates" the client when executing business service requests.

Step 7: Transfer Domains to Production Environment

Decide which business services you want to make available in the production environments and identify their domains. To export and import domains, use the Domain Transfer utilities in the Business Service Administration subsystem. When you use the utilities to copy the table data from one environment to another, the domains, business service definitions and, optionally, the step library chains are also copied.

> **Note:** For information on importing and exporting domains, see **Transfer Domains, Business Services, and Step Library Chains**.

Step 8: Transfer Group and User Tables (Optional)

If users and groups have already been defined in the production environments, you can omit this step. Otherwise, use the Export Groups and Import Groups utilities to copy this data to your production environments.

If you use the Export and Import Groups utilities, the permissions that grant access between groups and domains are not transferred. You must define security definitions in the Business Service Administration subsystem for the business services you are distributing.

> **Tip:** To make a mirror image copy of the Natural Business Services system files, use ADAULD to unload and ADALOD to reload.

> **Notes:**

1. For information on exporting and importing groups, see **Transfer Groups and Associated Users**.
2. For information on defining security definitions, see **Setting Business Service Security Options**.

Data Transfer Utilities

The data transfer utilities allow you to copy domains (which include business services and step library chains) and groups (which include associated users) between one Natural Business Services system file and another. These utilities copy data to and from either a work file or a PC file, depending on the definition of the WORK parameter in your NATPARM.

This section describes how to use the data transfer utilities in the Business Service Administration subsystem. The following topics are covered:

- Transfer Domains, Business Services, and Step Library Chains
- Transfer Groups and Associated Users
- Use the Data Transfer Utilities in Batch Mode

Transfer Domains, Business Services, and Step Library Chains

Use the Export Domains and Import Domains utilities to transfer domains, business services, their associated Natural subprograms, and step library chains from one Administration subsystem file to another. These utilities copy the following:

- Domain definitions

- Definitions for all business services with which the domain is associated, such as the service descriptions, methods, and subprogram proxy names associated with the business service definition

- Natural subprograms associated with the business services (optional)

- Step library chains associated with the domain (optional when importing domains)

This section covers the following topics:

- Access the Domain Transfer Utilities
- Export Domains, Business Services, and Step Library Chains
- Import Domains, Business Services and Step Library Chains

Access the Domain Transfer Utilities

▶ **To access the domain transfer utilities:**

1   Enter "AA" in `Function` on the **Business Service Administration Subsystem** main menu.

    The **Application Administration** main menu is displayed.

2   Enter "MM" in `Function`.

    The **Application Administration Maintenance** menu is displayed.

3   Enter "DT" in `Function`.

The **Transfer Domains** menu is displayed. For example:

```
BS_XFER2    ***** Business Service Administration Subsystem *****    CDLAYMN1
Feb 14,08                  - Transfer Domains Menu -                  09:39 PM


                  Functions

                  ------------------------------------------------

                  XD    Export Domains

                  ID    Import Domains

















                  ?     Help

                  .     Terminate

                  ------------------------------------------------

Function .......... __




Command .......... _____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      help  retrn quit       flip                                       main
```

4   Enter the code for the utility you want to access in `Function`.

The window for that utility is displayed.

Export Domains, Business Services, and Step Library Chains

▶ **To export domains, business services, and step library chains:**

1   Enter "XD" in `Function` on the **Transfer Domains** menu.

The **Export Domains** window is displayed. For example:

```
BS_EXDOM                                                      BSEXDOM0
  Domains.............    From.......  _____
                         Thru........  _____
* Business Service....    _____
  Latest version......   _
  Associated Subprograms _   Dest. Library _____    XREF _
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11
      help  retrn quit        flip
```

💡   **Tip:** You can invoke the Export utility in batch mode by entering "BS_EXDOM" at the Next prompt.

2   Specify the following options:

Field	Description
Domains From Thru	Domain(s) from which to export data. ■ To export one domain, type the name of the domain in `From`. ■ To export a range of domains, type the name of the first domain in `From` and the name of the last domain in `Thru`. ■ To export all domains with the same initial characters, type the characters followed by an asterisk (*) in `From`. ■ To export all domains up to and including a specific domain, type the name of the domain in `Thru`.
Business service	Business service to export, along with the associated domain and step library chains. ■ To export a business service, type the name of the domain containing the service in `From` and the name of the business service in `Business service`. By default, all versions of the business service are exported. ■ To export all business services with the same initial characters from the specified domain, type the characters followed by an asterisk (*) in `Business service`. **Note:** Business services can only be exported from one domain at a time; you cannot use an asterisk (*) in `From` or specify a domain name in `Thru`.

Field	Description
	Tip: When exporting a business service, select the service from the help window. This ensures the correct spelling and case of the service name. In addition, the name of the domain is retrieved.
Latest version	To only export the business service(s) with the highest version number, select this field. **Note:** The latest version number does not necessarily indicate the one with the most recent timestamp.
Associated Subprograms	To export the Natural subprograms that implement the selected business services, select this field. **Note:** The Natural SYSOBJH utility is used internally to export the associated subprograms; you must have access to this utility to export these subprograms.
Dest. Library	To override the name of the destination library, type the new name in this field (by default, the source library name is used).
XREF	To export XREF information for each Natural object associated with the services, select this field.

3   Press Enter to export the selected data.

   If you are exporting to a PC file, the Download window is displayed to specify the file in which to load the data. You can repeat steps 2 and 3 as many times as required. The selected services are exported one after the other to work file 2 and the associated subprograms are exported to work file 1 (when Associated Subprograms is selected). A report will be routed to your logical printer 1.

   A report listing the unloaded associated subprograms is saved as a text member named NBSDPLOY in the WORKPLAN library in the FUSER file (in batch mode, the report is printed in SYSOUT or the console).

4   Enter "." in `From` or press PF2 (retrn) to terminate the export operation and return to the **Transfer Domains** menu.

   Note: After the data has been successfully exported, a confirmation message is displayed. If the operation was not successful, a message describing the problem is displayed.

Import Domains, Business Services and Step Library Chains

You can import any domain that has been exported using the Export Domains utility.

▶ **To import domains, business services, and step library chains:**

1   Enter "ID" in `Function` on the **Transfer Domains** menu.

The **Import Domains** window is displayed. For example:

```
BS_IMDOM                                                        BSIMDOM0


  Import Domains........ _     <OR>     Scan input file....... _


  Domain........ _____

* Bus. Service... _____


  Replace steplib chains _

  Associated Subprograms _  Replace option _  XREF _  Dest. Library _____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF1
      help  retrn quit        flip                                      mai
```

> **Tip:** You can invoke the Import utility in batch mode by entering "BS_IMDOM" at the
> prompt.

2   Specify the following options:

Field	Description
Import Domains	To import domain data to the current library, select this field.
Scan input file	To only create a report of the data to be imported, select this field. The report lists the steplib chains, domains, and interfaces (business service definitions) and is written to the default device for Report 1 as set up by your system administrator.
Domain	To import a single business service, enter the name of the domain containing the service in this field and the name of the business service in the Bus. Service field. Wildcard characters are not supported. The domain name is combined with the name in the Bus. Service field to create a "filter" that uniquely identifies a business service in the input workfile. The resulting filter can be used by either the Import Domains or Scan input file function.

Field	Description
	Note: If you select this field, you cannot select the Associated Subprograms field.
Bus. Service	Name of the business service to import or scan from the specified domain. Wildcard characters are not supported. **Note:** This field must be specified in combination with the Domain field.
Replace steplib chains	To not replace step library chains in the target environment if they have the same names as those being imported, select this field.
Associated Subprograms	To import the Natural subprograms that implement the selected business services, select this field. **Note:** 1. If you select this field, you cannot select the Domain/Bus. Service fields. 2. The Natural SYSOBJH utility is used internally to import the associated subprograms; you must have access to this utility to import the subprograms.
Replace option	To replace existing data with imported data, select this field. **Note:** To select this option, you must provide valid input for the SYSOBJH utility (for example, " ", "A", "E", or "O"). For more information, see the Natural Utilities documentation.
XREF	To import XREF information for each Natural object associated with the services, select this field. **Note:** To select this option, you must provide valid input for the SYSOBJH utility (for example, " ", "N", "X", "Y", "F", or "D"). For more information, see the Natural Utilities documentation.
Dest. Library	To override the name of the destination library, type the new name in this field (by default, the source library name is used).

3 Press Enter.

If your NATPARM specifies a PC file, the Upload Data window is displayed to specify the file to be uploaded or scanned. If your NATPARM specifies a work file, the import utility uses that work file. Business services are imported from work file 2 and the associated subprograms are imported from work file 1 (when Associated Subprograms is selected). A report will be routed to your logical printer 1.

A report listing the associated Natural subprograms (previously unloaded using the Export utility) is saved as a text member named NBSDPLOY in the WORKPLAN library in the FUSER file (in batch mode, the report is printed in SYSOUT or the console).

Notes:

1. After the data has been successfully imported or scanned, a confirmation message is displayed. If the operation was not successful, a message describing the problem is displayed.

2. The business service data is not synchronously loaded with the associated subprograms. If the execution is cancelled, the loaded business service data may be out of synch with the subprogram data. If this occurs, restart the import process from the beginning.

3. The option to create a filter using the Domain and Bus. Service names does not affect the loading or scanning of steplib records, since these are independent objects. Regardless of the filter, all steplib records will be loaded or scanned.

Transfer Groups and Associated Users

Use the Export Groups and Import Groups utilities to copy groups and their associated users from one Business Service Administration subsystem file to another. This section covers the following topics:

- Access the Group Transfer Utilities
- Export Groups and Associated Users
- Import Groups and Associated Users

Access the Group Transfer Utilities

▶ **To access the group transfer utilities:**

1   Enter "SA" in `Function` on the **Business Service Administration Subsystem** main menu.

     The **System Administration** main menu is displayed.

2   Enter "MM" in `Function`.

     The **System Administration Maintenance** menu is displayed.

3   Enter "DT" in `Function`.

     The **Transfer Groups** menu is displayed. For example:

```
BS_XFER    ***** Business Service Administration Subsystem *****    CDLAYMN1
Feb 14,06                   - Transfer Groups Menu -                09:42 PM


                  Functions


                  ------------------------------------------------


                  XG    Export Groups


                  IG    Import Groups
```
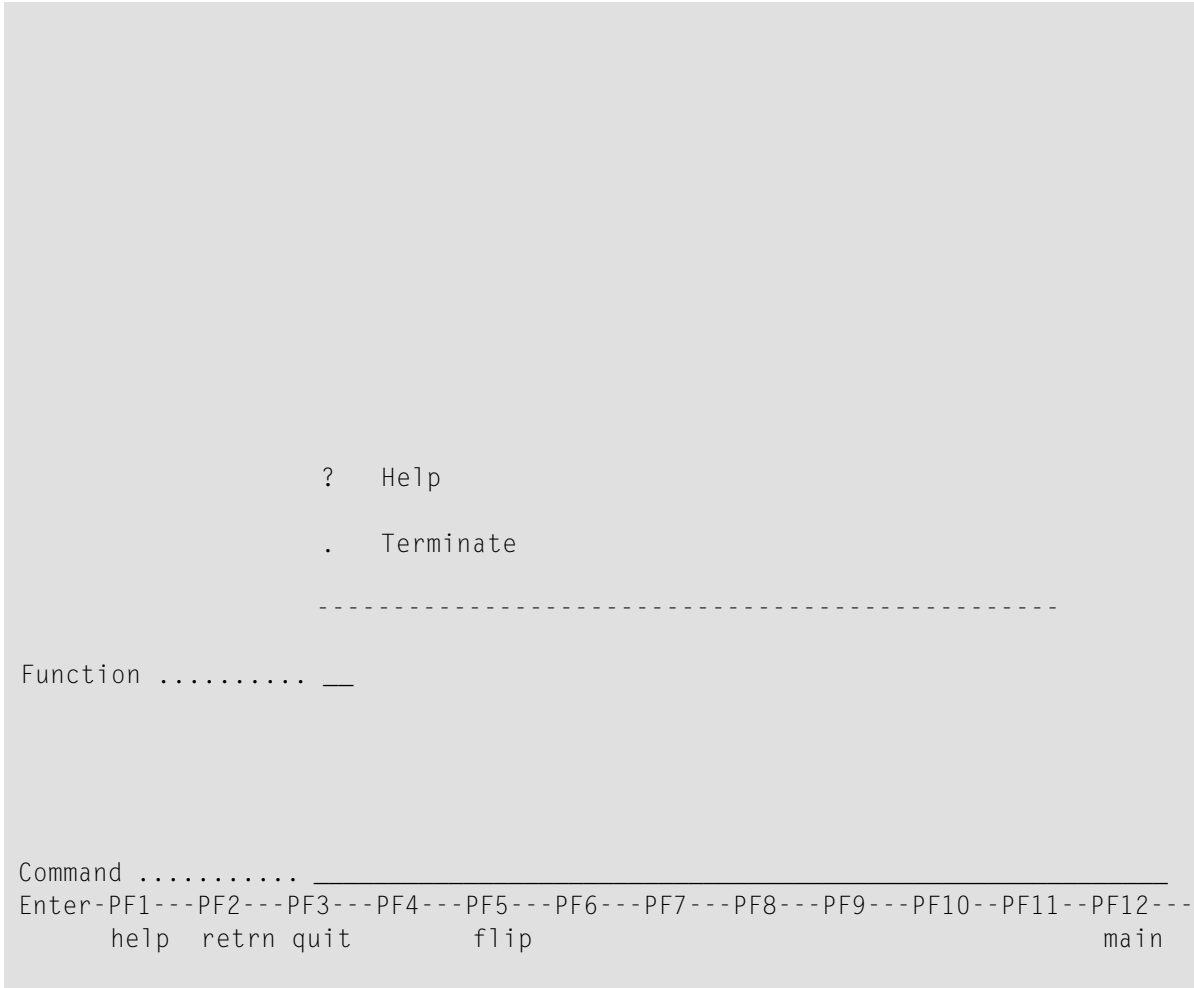
```
                           ?    Help

                           .    Terminate

                           ----------------------------------------------
Function .......... __




Command ........... _____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
       help  retrn quit        flip                                    main
```

4   Enter the code for the utility you want to access in Function.

The window for that utility is displayed.

Export Groups and Associated Users

▶ **To export groups and associated users:**

1   Enter "XG" in Function on the **Transfer Groups** menu.

The **Export Groups** window is displayed. For example:

```
                        Export Groups
  BS_EXGRP                                      BSEXGRPO


            Groups
              From........ _____
              Thru........ _____
  Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9--
        help  retrn quit        flip
```

2   Specify the following options:

Field	Description
Groups	Group(s) from which to export data. Wildcard characters are supported.
From Thru	■ To export one group, type the name of the group in `From`. ■ To export a range of groups, type the name of the first group in `From` and the name of the last group in `Thru`. ■ To export all groups up to and including a specific group, type the name of the group in `Thru`.

3   Press Enter to start the export process.

   If you are exporting to a PC file, the Download window is displayed to specify the file into which the data is to be transferred.

After the data is exported, a confirmation message is displayed. If the operation was not successful, a message describing the problem is displayed.

Import Groups and Associated Users

You can import any group that has been exported using the Export Groups utility.

▶ **To import groups and associated users:**

1   Log onto the library to which you want to import the groups.

2   Invoke the Administration subsystem.

3   Access the **Transfer Groups** menu.

4   Enter "IG" in `Function` on the **Transfer Groups** menu.

The **Import Groups** window is displayed. For example:

```
                        Import Groups
  BS_IMGRP                                               BSIMGRP0


        Import groups...........................: _

        Scan input file.........................: _
  Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---
        help  retrn quit         flip
```

5   Specify the following options:

Field	Description
Import groups	To import group data to the current library, select this field.
Scan input file	To only create a report of the data to be imported, select this field. The report lists the groups and users in the data file and is written to the default device for Report 1 as set up by your system administrator.

6   Press Enter.

If your NATPARM specifies a PC file, the Upload Data window is displayed to locate and select the file to be imported or scanned. If your NATPARM specifies a work file, the import utility uses the data from that work file.

After the data is imported or scanned, a confirmation message is displayed. If the operation was not successful, a message describing the problem is displayed.

Use the Data Transfer Utilities in Batch Mode

▶ **To invoke a data transfer utility in batch mode:**

■   Enter one of the following commands at the Next prompt:

Utility	Command
Export Domains	BS_EXDOM
Import Domains	BS_IMDOM
Export Groups	BS_EXGRP
Import Groups	BS_IMGRP

The following JCL examples call the data transfer utilities in batch mode:

- Export Domains Example

- Import Domains Example
- Import Business Service Example
- Export Groups Example
- Import Groups Example

Export Domains Example

```
. . .
//CMPRT01 DD SYSOUT=X
//CMWKF01 DD DSN=DEV.DOMAINS.TEMP,DISP=(,CATLG),UNIT=SYSDA,
//         DCB=(RECFM=VB,LRECL=4624,BLKSIZE=4628),
//         SPACE=(CYL,(5,2),RLSE)
//SYSIN    DD *
GLOBALS IM=D
LOGON SYSBIZ
BS_EXDOM <from domain> <to domain>
```

Import Domains Example

```
. . .
//DDCARD DD *
ADARUN MODE=MULTI,DBID=019,PROG=USER,SVC=247
//CMPRT01 DD SYSOUT=X
//CMWKF01 DD DSN=DEV.DOMAINS.TEMP,DISP=SHR
//CMSYNIN DD *
//SYSIN   DD *
GLOBALS IM=D
LOGON SYSBIZ
BS_IMDOM X,,X  << --- use this to replace steplibs
BS_IMDOM X,,,  << --- use this to not replace steplibs
BS_IMDOM X,X,  << --- use this to only scan the import file
```

> **Note:** If you mark the Replace Steplibs and Scan Import file options, the report indicates which steplibs were replaced.

Import Business Service Example

```
//SYSOUT    DD SYSOUT=&SYSOUT
//*
//CMWKF01  DD DSN=EXPRT.SUBPGM,DISP=SHR
//CMWKF02  DD DSN=EXPRT.DOMAINS,DISP=SHR
//*
//CMPRINT   DD SYSOUT=X
//CMPRT01   DD SYSOUT=X
//*
//CMSYNIN   DD *
LOGON SYSBIZ
BS_IMDOM
X,,MYDOMAIN,MYSERVICE,X,,,Y
.
```

```
FIN
/*
```

Export Groups Example

```
. . .
//CMPRT01 DD SYSOUT=X
//CMWKF01 DD DSN=DEV.DOMAINS.TEMP,DISP=(,CATLG),UNIT=SYSDA,
//           DCB=(RECFM=VB,LRECL=4624,BLKSIZE=4628),
//           SPACE=(CYL,(5,2),RLSE)
//SYSIN    DD *
GLOBALS IM=D
LOGON SYSBIZ
BS_EXGRP <from group> <to group>
```

Import Groups Example

```
. . .
//DDCARD DD *
ADARUN MODE=MULTI,DBID=019,PROG=USER,SVC=247
//CMPRT01 DD SYSOUT=X
//CMWKF01 DD DSN=DEV.GROUPS.TEMP,DISP=SHR
//CMSYNIN DD *
//SYSIN   DD *
GLOBALS IM=D
LOGON SYSBIZ
BS_IMGRP X,, << --- use this to import groups
BS_IMGRP,X,  << --- use this to only scan the import file
```

Layout of the Export Users and Groups Work File

To import table data from external systems, such as an in-house security system, you can create your own export utility. This utility will copy your data into a transfer file that conforms to the import file specifications expected by the Business Service Import Groups utility. To help create this module, Natural Business Services supplies the BSSI_EX1 module in the SYSBIZ library as a sample template for a program that uses the work file definitions.

The work file is configured into variable length record types, with each record consisting of a line in an ASCII file. Lines are separated by CR+LF characters. The following types of records are supported:

- Header record
- Group record
- User record

Each record is identified by a record ID. The information contained in each record must strictly follow the guidelines outlined in the following tables with regard to data values and relative positioning within the record.

Header Record

This record must be the first record in the work file. It identifies the version of the utility that created the file and the version of the Natural Business Services system file from which data was exported. The Header record fields are:

Field	Value	Length	Start Position	End Position	Description
Record Type	<H	2	1	2	Start of Header record
Utility Version	5.5.1	10	3	12	Version of export utility
Data Version	0551	4	13	16	Version of Natural Business Services system file
Reserved		139	17	155	Reserved for future use
Record Type	H>	2	156	157	End of Header record

Group Record

Group records contain the information found in the Natural Business Services Groups table. The Group record fields are:

Field	Value	Length	Start Position	End Position	Description
Record Type	<G	2	1	2	Start of Group record
Group ID		8	3	10	Unique Group ID
Group Name		50	11	60	Descriptive group name
Record Type	G>	2	61	62	End of Group record

User Record

User records contain the information found in the Natural Business Services Users table. When creating an export file for both groups and users, ensure that Group records are copied to the work file before User records.

> **Note:** The Groups Export utility copies User records to the work file in an encrypted format.
>
> If you write your own export utility to create an export work file, passwords cannot be encrypted because the Import Groups utility cannot decrypt the password before updating the User table.

The User record fields are:

Field	Value	Length	Start Position	End Position	Description
Record type	<U	2	1	2	Start of User record
User ID		8	3	10	Unique user ID
Password		40	11	50	User password
User name		50	51	100	User name
Debug library		8	101	108	Name of valid Natural library
Debug module derive method	T or U	1	109	110	Derive method for debug text members: timestamp (T) or user ID (U)
User language	1-60	3	111	113	Valid Natural *Language value, zero-filled, right-justified
Linked group count	1-10	3	114	116	Number of groups user is linked to, zero-filled, right-justified (must correspond to the number of entries in the following field)
Linked group IDs		80	117	196	Each Group ID written to the work file in a previous group record, redefined into 10 occurrences of eight bytes each, left-justified within eight bytes
Record type	U>	2	197	198	End of User record

13 Customizing Natural Business Services

This section describes how to customize the default behavior of Natural Business Services. The following topics are covered:

Modify/Test the Naming Conventions for Natural Objects

This section describes how the Natural Business Services wizards create names for the Natural objects they generate and how to include your own naming conventions.

Since one Natural Business Services wizard may generate several different Natural objects, each type of object must be uniquely identified. For example, the Business Service wizard creates at least one subprogram that handles logic and one subprogram proxy.

The different styles and their unique identifiers are stored in an LDA called WTRNAMEL. Natural Business Services names the modules as follows:

■ One character for the object style and one character for the category

   To change these characters, modify WTRNAMCL and then recompile CTENAMEL and WTR-NAME.

■ Five characters supplied by you

■ Optionally, a number to help retain uniqueness within a library

How the unique identifier is used, where it is used, and how unique Natural object names are derived is handled in WTRNAME. To change the location of the category, style, and/or five characters, modify WTRNAME and then test the generation of the Natural module names.

▶ **To test the generation of Natural module names:**

1   Enter the "menu D A GU" direct command.

   The General Utility Subprograms menu is displayed. For example:

```
CTEMENGU                    ***** Natural Construct *****                 CTEMNMO
May 26                    General Utility Subprograms Menu                 1 of 1



                      Functions

                      ----------------------------------------------

                      SR Storage required for a variable

                      BN Build Natural format

                      CT Convert text to upper, lower, mixed
```

```
                          FP Find PF-key related information

                          SL Shorten a long variable name

                          GN Get a NATPARM parameter value

                          CS Determine seconds from given start time

                          NA Test NBS naming conventions




                          ?  Help

                          .  Return

                          ---------------------------------------------

Function ........... __

Command ............ _____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      help  retrn quit                                              lang
```

2   Enter "NA" in Function on the General Utility Subprograms menu.

The following screen is displayed. For example:

```
Up to 5 characters TEST_



(blank searches *LIBRARY, 1 searches provided library,2 searches both)

Provided library: _____   _ Library search requirement:



When a subprogram proxy client is generated,

the name is derived from the server subprogram proxy name.

Since this name is larger than 5 characters, the Name field

is used for input and output. To obtain the client proxy name,
```

```
enter the server proxy name in the Name field.



Name: _____
```

By default, TEST is used for the example. Change the default characters and specify the name of the library (if not the current library).

- To search the current library, leave the Provided library and Library search requirement fields blank.

- To search the Provided library, type the library name in Provided library and type "1" in the Library search requirement field.

- To search both libraries, type the library name in Provided library and type "2" in the Library search requirement field.

3 Enter the name of the server proxy in Name.

Natural Business Services displays the default name of the client proxy. For example, if you entered "MYPROXY" in Name, the following is displayed:

```
 07-05-26            - Driver for subprogram WTRNAME -              17:45:40
      You can customize WTRNAMCL, along with WTRNAME, for site-specific

                        naming conventions.

              Names within a category must be UNIQUE

      as well as names ACROSS all object maint and object browse modules




Given a server proxy name of:  MYPROXY

The client proxy name will be: MYPROXC
```

4 Press Enter.

The following screen is displayed:

```
 07-05-26           - Driver for subprogram WTRNAME -                17:49:37
       You can customize WTRNAMCL, along with WTRNAME, for site-specific

                             naming conventions.

                  Names within a category must be UNIQUE

      as well as names ACROSS all object maint and object browse modules




Object Module                    Name based on 5 char TEST

------------------------------  -----------------------

CATEGORY OBJECT-BROWSE-SELECT

OBJECT-BROWSE-SELECT             TESTNB1

SUBPROGRAM-PROXY                 TESTNB1Y

SUBPROGRAM-PROXY-CLIENT          TESTNB1C


CATEGORY OBJECT-BROWSE

OBJECT-BROWSE-ROW-PDA            TESTDA1

OBJECT-BROWSE-KEY-PDA            TESTKA1

OBJECT-BROWSE-RESTRICTED-PDA     TESTPA1

OBJECT-BROWSE-STATIC-LDA         TESTLA1

OBJECT-BROWSE-SUBP               TESTNA1

SUBPROGRAM-PROXY                 TESTNA1Y

SUBPROGRAM-PROXY-CLIENT          TESTNA1C
```

This screen displays the default names for the different models.

5   Press Enter again.

The following screen is displayed:

```
 07-05-26            - Driver for subprogram WTRNAME -              18:01:28
        You can customize WTRNAMCL, along with WTRNAME, for site-specific

                            naming conventions.

                   Names within a category must be UNIQUE

       as well as names ACROSS all object maint and object browse modules




CATEGORY OBJECT-MAINT

OBJECT-MAINT-DATA-PDA           TESTDM1

OBJECT-MAINT-RESTRICTED-PDA     TESTPM1

OBJECT-LDA                      TESTLM1

OBJECT-MAINT-SUBP               TESTNM1

SUBPROGRAM-PROXY                TESTNM1Y

SUBPROGRAM-PROXY-CLIENT         TESTNM1C



CATEGORY OBJECT-GENERIC

OBJECT-GENERIC-SUBP             TESTNG1

SUBPROGRAM-PROXY                TESTNG1Y

SUBPROGRAM-PROXY-CLIENT         TESTNG1C
```

6   Press Enter again.

The following screen is displayed:

```
 07-05-26            - Driver for subprogram WTRNAME -              18:02:22
      You can customize WTRNAMCL, along with WTRNAME, for site-specific

                            naming conventions.

                    Names within a category must be UNIQUE

      as well as names ACROSS all object maint and object browse modules




 CATEGORY TRANSFORM-BROWSE

 TRANSFORM-BROWSE                    TESTOT

 SUBPROGRAM-PROXY                    TESTOTY

 SUBPROGRAM-PROXY-CLIENT             TESTOTC
```

7   Press Enter again.

The following screen is displayed, showing any conflicts in the naming conventions:

```
 07-05-26            - Driver for subprogram WTRNAME -              18:03:34
      You can customize WTRNAMCL, along with WTRNAME, for site-specific

                            naming conventions.

                    Names within a category must be UNIQUE

      as well as names ACROSS all object maint and object browse modules




    Unique Name requirements      Name       Conflict

    -----------------------       ----       --------

  1 OBJECT-BROWSE-SELECT          TESTNB1

  2 SUBPROGRAM-PROXY              TESTNB1Y

  3 SUBPROGRAM-PROXY-CLIENT       TESTNB1C
```

```
 5 OBJECT-BROWSE-ROW-PDA       TESTDA1

 6 OBJECT-BROWSE-KEY-PDA       TESTKA1

 7 OBJECT-BROWSE-RESTRICTED-PDA TESTPA1

 8 OBJECT-BROWSE-STATIC-LDA    TESTLA1

 9 OBJECT-BROWSE-SUBP          TESTNA1

10 SUBPROGRAM-PROXY            TESTNA1Y

11 SUBPROGRAM-PROXY-CLIENT     TESTNA1C

13 OBJECT-MAINT-DATA-PDA       TESTDM1

14 OBJECT-MAINT-RESTRICTED-PDA TESTPM1

15 OBJECT-LDA                  TESTLM1

16 OBJECT-MAINT-SUBP           TESTNM1
```

8   Press Enter again.

The last screen is displayed:

```
07-05-26           - Driver for subprogram WTRNAME -              18:20:04
     You can customize WTRNAMCL, along with WTRNAME, for site-specific

                         naming conventions.

            Names within a category must be UNIQUE

   as well as names ACROSS all object maint and object browse modules




17 SUBPROGRAM-PROXY            TESTNM1Y

18 SUBPROGRAM-PROXY-CLIENT     TESTNM1C
```

Define Default Specification Parameters

You can define default parameter values for the model specification panels in Natural Construct. The supplied models retrieve many of the default values by issuing a CALLNAT to the CSUDEFLT subprogram. Prior to returning the defaults, CSUDEFLT checks to see whether the values have been overridden by the user-defined CSXDEFLT sample exit subprogram. If so, the overridden values are returned to the model. When the Business Service wizard generates the models to create business services, your default values will be used.

The CSXDEFLT sample exit subprogram provides user-defined default values for model parameters that can be overridden on the specification panels, as well as internal model parameters that are not displayed on the panels.

> **Note:** For more information, see Define Default Specifications.

Modify Logon Security

The CSXSECX sample exit subprogram allows you to take advantage of Natural Security LOGON user exit. If the security routines in CSXSECX are defined and the object code is in the SYSLIBS library, they are processed before the standard Natural Business Services security routines are processed.

The parameters for CSXSECX are:

```
1 STEPLIB-DATA    (A144)
1 REDEFINE STEPLIB-DATA
  2 STEPLIB-DATA-STRUCT  (1:8)
   3 LIB-NAME (A8)
   3 LIB-DBID (N5)
   3 LIB-FNR (N5)
1 PROXY-SUBPROGRAM (A8)
1 USER-ID (A32)
1 REDEFINE USER-ID
 2 USER-ID-A8 (A8)
1 DOMAIN (A8)    /* Domain name
1 BUSINESS-SERVICE (A32) /* business service name
1 BUSINESS-SERVICE-VERSION (N6) /* business service version
1 BUSINESS-METHOD (A32) /* business service version
```

> **Note:** Although the parameters above are accessible within this exit, you cannot change the parameter values here; you can only add additional security checks.

▶ **To customize CSXSECX:**

1   Copy the subprogram from the SYSCSTX library to the SYSCST library.

2   Add your own security settings.

3   Stow the subprogram.

4   Move the object code to the SYSLIBS library to make the routine active.

When the dispatcher is started, it checks to see if the object code for CSXSECX is in the SYSLIBS library. If the object code is not in the SYSLIBS library, the dispatcher assumes there is no customized security checking and will process the standard Natural Business Service security routines. If the object code is in the SYSLIBS library, the dispatcher will execute this code first and then process the standard security routines.

For information on the sample exit subprograms, see Access and Use the Sample Exit Subprograms.

> **Note:** You can also enhance the standard Natural Business Services security routines. For information, see **Setting Business Service Security Options**.

Modify the BACKOUT or END TRANSACTION Processing

The CSXTRANS sample exit subprogram allows you to issue BACKOUT or END TRANSACTION statements. If the object code for CSXTRANS is in the SYSLIBS library, it is executed after the call to a business service has been made.

> **Note:** An Object Maint-generated subprogram issues an END TRANSACTION statement when CDAOBJ2. #ET-IF-SUCCESSFUL is True. Therefore, a BACKOUT TRANSACTION statement may not be issued when expected.

The parameters for CSXTRANS are:

```
1 STEPLIB-DATA (A144)
1 REDEFINE STEPLIB-DATA
 2 STEPLIB-DATA-STRUCT (1:8)
  3 LIB-NAME (A8)
  3 LIB-DBID (N5)
  3 LIB-FNR (N5)
1 PROXY-SUBPROGRAM (A8)
1 USER-ID (A32)
1 REDEFINE USER-ID
 2 USER-ID-A8 (A8)
1 DOMAIN (A8) /* Domain name
1 BUSINESS-SERVICE (A32) /* business service name
1 BUSINESS-SERVICE-VERSION (N6) /* business service version
1 BUSINESS-METHOD (A32) /* business service version
```

```
1 START-OF-PROXY-TIMESTAMP (B8) /* time CALLNAT to business service was issued
**SAG END-EXIT
PARAMETER USING CDPDA-M
```

> **Note:** Changing any value except the values in CDPDA-M will have no effect; the values are reset to the values they were before entering this exit.

▶ **To customize CSXTRANS:**

1   Copy the subprogram from the SYSCSTX library to the SYSCST library.

2   Add your own BACKOUT or END TRANSACTION logic.

3   Stow the subprogram.

4   Move the object code to the SYSLIBS library to make the routine active.

When the dispatcher is started, it checks to see if the object code for CSXTRANS is in the SYSLIBS library. If the object code is not in the SYSLIBS library, the dispatcher assumes there is no customized END or BACKOUT TRANSACTION processing and will process the standard Natural Business Service security routines. If the object code is in the SYSLIBS library, the dispatcher will execute this code first and then process the standard security routines.

For information on the sample exit subprograms, see Access and Use the Sample Exit Subprograms.

> **Note:** You can also enhance the standard Natural Business Services security routines. For information, see **Setting Business Service Security Options**.

Modify the Supplied Generation Models

You can customize the business services generated by the Business Service wizard by modifying the Natural Construct generation models. You can modify the code frames, subprograms, and copycode used by the supplied models, as well as the external data areas and subprograms.

> **Note:** For more information, see Modifying the Supplied Models.

14 Using Custom Models with Natural Business Services

When using the Business Service wizard to generate services, you can select Natural Construct models you have created. This section describes how to set up the custom model for use with the Business Service wizard.

> **Note:** The custom model must generate a Natural subprogram.

▶ **To set up the custom model:**

1 Create a subprogram and PDA that Natural Business Services will use when the Business Service wizard is called.

This subprogram associates fields in the wizard to fields in the model parameter data area (PDA).

> **Note:** For an example of a subprogram and data area, refer to WTOBR2 and WTIOG2DT in the SYSCST library. The example is based on a custom model created from the Object-Maint-Subp model.

2 Add an entry in the models.xml file for the plug-in.

The models.xml file allows you to define which custom models are available in the wizard. For more information, refer to the comments and example in the *models.xml* file. This file is located in the NBS Eclipse plug-in subfolder or the installation folder (for the Natural plug-in).

The custom model will now be available for selection when a user selects the **By using custom code generation model** option on the first Business Service wizard panel. The model fields are displayed in a grid and the user can enter the required values to create the service.

A Appendix A: Parsing Server Startup Parameters

Natural Business Services supplies the BSSPARMN routine, which parses the keywords specified in the Server Start parameters fields on the Maintain Servers panel (for information, see **Define Server Start Parameters and Routines**). BSSPARMN uses the BSSPARMA and CDPDA-M parameter data areas, which define the interface to the routine.

This section describes these data areas. The following topics are covered:

BSSPARMA Data Area

This data area defines the parameter string, as well as the valid keywords. It returns the individual parameters after they have been parsed. The following input and output parameters are defined in this data area.

Input Parameters

The following table describes the input parameters for BSSPARMA:

Input Parameter	Description
PARM (A1/1:500)	String that defines the parameter values entered in the User parameters fields. If the value contains special characters, enclose it within quotes.
	On the input panel, this string is divided into five blocks of 50 characters. Normally, it contains a series of keyword=value combinations separated by commas (or alternate Natural input delimiter characters).
C#KEYWORD (I2)	Variable that tells BSSPARMN how many valid keywords are supplied in the KEYWORD field.
KEYWORD (A32/1:15)	Array containing valid keywords. For every valid keyword, specify an entry in this array. Also assign C#KEYWORD to indicate the number of keywords passed.
SHOW-RESULTS (L)	If this flag is set, BSSPARMN writes out the individual keyword values to the panel after parsing them. This allows you to supply the name of a test or trace facility.

Output Parameters

The following table describes the output parameters for BSSPARMA:

Output Parameter	Description
C#INDIVIDUAL-PARMS (I2)	Value indicating the number of individual parameter values identified by BSSPARMN.
KEYWORD-ENTERED (L/1:15)	This field is set to true for occurrence n, if the nth keyword specified in the KEYWORD array was followed by an input assign character. **Note:** The assign character is normally an equal sign (=), but it can be overridden by the Natural IA parameter.
VALUE (A200)	Field containing the value of a parameter. If the corresponding KEYWORD-ENTERED flag is set, this value corresponds to the KEYWORD value in the same occurrence. If KEYWORD-ENTERED is false, there may still be a value in the corresponding VALUE field, but this value was not preceded by a keyword.

CDPDA-M Parameter Data Area

If BSSPARMN was not able to parse the input parameters, CDPDA-M contains a message number in CDPDA-M.##MSG-NR to indicate the reason. This message number corresponds to a SYSERR message for the SYSBIZ library. Substitution parameters may also be returned in CDPDA-M.##MSG-DATA(1).

Example of Using BSSPARMN

```
DEFINE DATA
  LOCAL USING BSSPARMA
  LOCAL USING CDPDA-M
  LOCAL
  01 #I(I1)
END-DEFINE
*
* Parameter values in PARM-CHUNK are normally entered as INPUT
* values
BSSPARMA.PARM-CHUNK(1) :=
 'JOB-NAME=SOMEJOB,NATPARM="FNAT=(1,2),PROFILE=SYSBIZD"'
*
* define 3 valid parameters
BSSPARMA.C#KEYWORD := 3
BSSPARMA.KEYWORD(1) := 'NATPARM'
BSSPARMA.KEYWORD(2) := 'USER'
BSSPARMA.KEYWORD(3) := 'JOB-NAME'
*
```

```
* Call BSSPARMN to parse the parameters
CALLNAT 'BSSPARMN' BSSPARMA CDPDA-M
*
* If there were no errors, show the results
IF CDPDA-M.##RETURN-CODE = ' ' THEN
  FOR #I = 1 TO BSSPARMA.C#INDIVIDUAL-PARMS
    DECIDE FOR FIRST CONDITION
      WHEN BSSPARMA.KEYWORD-ENTERED(#I)
        PRINT BSSPARMA.KEYWORD(#I) 'equals'
        BSSPARMA.VALUE(#I)(AL=50)
      WHEN BSSPARMA.VALUE(#I) NE ' '
        PRINT 'Value without keyword' BSSPARMA.VALUE(#I)(AL=50)
      WHEN NONE
        IGNORE
    END-DECIDE
    IF NOT BSSPARMA.KEYWORD-ENTERED(#I) AND #I LE BSSPARMA.C#KEYWORD
    THEN
      PRINT 'No value entered for parameter' BSSPARMA.KEYWORD(#I)
    END-IF
  END-FOR
ELSE
  WRITE 'Parameter error' CDPDA-M.##MSG-NR CDPDA-M.##MSG-DATA(1)
END-IF
END
```

B   Appendix B: Additional Error Handling

To enhance the core error processing, Natural Business Services supplies the SPSERRN2 user exit. This user exit allows you to add special error processing, such as sending notification of errors to the operator.

▶ **To create and activate custom error processing:**

1   Edit the user exit in the SYSBIZ library.

2   Copy the object code for the modified user exit to the SYSLIBS library.

📄   **Notes:**

1. For backwards compatibility, the SPSERRN user exit is also supplied. Use SPSERRN2 for all new development.

2. If you are using the Natural Development Server (NDV), avoid using screen input/output parameters.

3. Do not stop or terminate processing in the subprograms or associated modules. To ensure complete tracking of internal business service information, control should return to the Natural Business Services error handler.

4. If you add custom error processing in either user exit, back up the program before upgrading Natural Business Services in the future. You can move the program back after installation is complete.

SPSERRN2 User Exit

The SPSERRN2 user exit uses the CDERRLD2 parameter data area, which contains the following parameters:

▪ MSG_TYPE

▪ ERROR_TYPE

▪ MSG_CATEGORY

▪ ERROR_CLASS

▪ ERROR_LIBRARY

▪ ERROR_PROGRAM

▪ ERROR_NUMBER

▪ ERROR_LINE

▪ ERROR_LEVEL_NBR

▪ ERROR_MSG

▪ NBS_SERVICE

- NBS_DOMAIN
- NBS_METHOD
- PROCESS_DESCRIPTION
- ERROR_DATE
- ERROR_TIME
- ERROR_TIMESTAMP

SPSERRN User Exit

The SPSERRN user exit is supplied for backwards compatibility. You should use the SPSERRN2 user exit for all new development.

SPSERRN uses the CDERRLDA parameter data area, which contains the following parameters:

- ERROR-NR
- ERROR-LINE-NR
- ERROR-STATUS
- ERROR-PROGRAM
- ERROR-PROGRAM-LEVEL

C   Appendix C: Utilities

This section describes the utility subprograms supplied with the Business Service Administration subsystem. To invoke these subprograms, you must be in the SYSBIZ library.

The following topics are covered:

Response Subprogram

The SPUREPLY subprogram is mainly used by servers to send responses back to a client. The response can be defined as a SYSERR message or a hardcoded text string.

This section covers the following topics:

- Features and Benefits
- Response Length Limitation
- Supported Methods
- Message Protocol
- Call Interface

Features and Benefits

SPUREPLY has the following benefits and features:

- Defines a standard protocol for exchanging messages.

- Enables messages to be multilingual if you define them in SYSERR.

- Performs message substitution of :1::2::3: within SYSERR messages.

- Can send other information in addition to a message.

Response Length Limitation

The maximum supported response length is 5000 bytes.

Supported Methods

SPUREPLY supports the following methods (defined in SPLREPLY). One of these methods must be assigned to the SPAREPLY.METHOD parameter before calling SPUREPLY:

Method	Description
SEND-REPLY	Sends a single message reply, with the End of Conversation option.
SEND-WITHOUT-EOC	Sends a multi-part reply. Use the SEND-REPLY method to send the last message of the reply.
LOOKUP-MESSAGE	Looks up the error message text, but does not send it.
SEND-MESSAGE-ONLY	Sends the message text without the standard protocol information.
SEND-MESSAGE-ONLY-WITHOUT-EOC	Same as SEND-MESSAGE-ONLY, but does not include End of Conversation option.

Message Protocol

All messages sent to the client use the following protocol:

Message	Protocol
SIGNATURE (A6)	MSG111 constant. Defines the structure of the send buffer.
RESPONSE-CODE (N4)	Response code passed to SPUREPLY in SPAREPLY. Successful responses use a response code of zero. Other predefined response codes are: ■ 001 (Replica ID was not matched) ■ 9999 (Natural runtime error)
REPLICA-ID (A32)	Replica ID passed from SPAETB.
BUSINESS-SERVICE (A32)	Passed from SPAETB.
SYSERR-LIBRARY (A8)	Name of the SYSERR library containing the message.
MSG-NR (N4)	Number of the SYSERR message.
MESSAGE (A1/1:V)	Message area of send buffer. In most cases, this area contains a message looked up in SYSERR by SPUREPLY. Additional information can also be passed in this area.

Call Interface

SPUREPLY supports the following interface:

```
PARAMETER USING SPAREPLY /* Specific parameters
PARAMETER                         /* The message portion of the send buffer
  01 SPAREPM
    02 INPUT-OUTPUTS
      03 BUFFER-LENGTH (I2)
      03 MSG-BUFFER (A1/1:V)
PARAMETER USING SPAETB      /* Parameters to SPUETB
PARAMETER USING ETBCB       /* Standard Broker control block
PARAMETER USING CDPDA-M     /* Standard message area
```

This section covers the following topics:

- SPAREPLY Data Area
- SPAREPM Data Area

SPAREPLY Data Area

This data area is passed to SPUREPLY. It contains the following data:

```
Parameter SPAREPLY  Library S452                          DBID 13000 FNR  1001
Command                                                                  > +
I T L Name                          F Leng Index/Init/EM/Name/Comment
Top - ----------------------------- - ---- --------------------------------
  *
  *
  *   Data Area Name: SPAREPLY                          Function
  *   Created on....: Jun 12, 98                        ========
  *   Created by....: SAG                    This data area is passed to
  *                                          SPUREPLY which is used to
  *                                          send a reply back to a client.
  *
  *                                          The reply structure is
  *                                          defined in SPLREP.
    1 SPAREPLY
    2 INPUTS
    3 METHOD                        I    1 /* See SPLREPLY
    3 RESPONSE-CODE                 N    4 /* This response, use zero for
  *                                          /* successful response.
    3 SYSERR-INFO
    4 MSG-NR                        N    4 /* SYSERR Message number
    4 SYSERR-LIBRARY               A    8 /* Defaults to SYSBIZ
    4 MSG-DATA                      A   32 (1:3) /* Subs. values
  *                                          /* May contain *NNNN references
    3 TRANSLATE                     L      /* Translate character set. If
  *                                          /* currently EBCDIC, message
  *                                          /* will be translated to ASCII
```

```
*                                            /* and vise-versa.
  3 EMBEDDED-MSG-INFO                         /* This structure is only used
*                                            /* when the message to be looked
*                                            /* up is only a portion of the
*                                            /* data to be sent. In this case
*                                            /* you must indicate where the
*                                            /* message is in the send buffer
  4 MSG-START                     I    2 /* Byte location of start of msg
  4 MSG-LENGTH                    I    2 /* Total length of message
*                                            /* portion.
```

The fields in this data area include:

Field	Description
METHOD (I1)	Indicates whether you want to perform a send with EOC, send without EOC, or just look up the message text. Assign a value from SPLREPLY.
RESPONSE-CODE (N4)	Contains the response code value sent to the client.
MSG-NR (N4)	If a message is looked up in SYSERR, contains the message number.
SYSERR-LIBRARY (A8)	Name of the library in which to look up messages. By default, all messages are looked up in the SYSBIZ library. If this is not true, specify the library name.
MSG-DATA (A32/1:3)	Contains up to three values for substitution into the message. These values replace the :1::2::3: placeholders in the SYSERR message. Substitution values can be looked up in SYSERR by specifying message data in *nnnn format.
TRANSLATE (L)	Indicates whether the message is translated (from EBCDIC to ASCII or vice versa).
EMBEDDED-MSG-INFO	Indicates whether the message is a portion of the data sent. You must specify where the message is in the Send buffer.
MSG-START (I2)	Indicates the starting position of the message portion of the send buffer if the message retrieved from SYSERR only represents a portion of the data to be sent.
MSG-LENGTH (I2)	Indicates the length of the message portion of the send buffer. This field is only required when MSG-START is assigned.

SPAREPM Data Area

This data area is an example of a standard message area that can be passed to SPUREPLY. Use SPAREPM to send messages up to 250 characters in length. After SYSERR messages are looked up, the resulting message text is returned in this parameter. The values in SPAREPLY.MSG-START and SPAREPLY.MSG-LENGTH determine where the message is assigned. If these values are zero, the message is returned, starting at position 1 and continuing to SPAREPM.BUFFER-LENGTH.

The SPAREPM data area contains the following fields:

```
Local     SPAREPM    Library S452                          DBID 13000 FNR   1001
Command                                                                       > +
I T L Name                              F Leng Index/Init/EM/Name/Comment
All - ------------------------------ - ---- --------------------------------
   *    Data Area Name: SPAREPM                          Function
   *    Created on....: Jun 12, 98                       ========
   *    Created by....: SAG                   This data area can be used as
   *                                          the second parameter to
   *                                          SPUREPLY. When a message number
   *                                          is passed to SPUREPLY, the
   *                                          message text is returned in
   *                                          this parameter.
   *                                          Alternatively, the message to
   *                                          be sent can be passed to
   *                                          SPUREPLY using this parameter.
     1 SPAREPM
     2 INPUT-OUTPUTS
     3 BUFFER-LENGTH                   I    2 INIT<250>
     3 MSG-BUFFER                      A    1 (1:250)
   R 3 MSG-BUFFER
     4 MSG-STRING                      A  250
```

To send information other than a standard message, copy SPAREPM and define the fields you want to send (up to 5000 bytes). To reflect the size of data to be sent, assign the BUFFER-LENGTH field.

Example of a Call

```
/*
/* SYSBIZ/1001: Invalid request:1:sent to:2:expecting:3:
ASSIGN SPAREPLY.MSG-NR = 1001
ASSIGN SPAREPLY.MSG-DATA(1) = #COMMAND
ASSIGN SPAREPLY.MSG-DATA(2) = *PROGRAM
ASSIGN SPAREPLY.MSG-DATA(3) = '''CREATE'''
ASSIGN SPAREPLY.RESPONSE-CODE = 1 /* Invalid command
PERFORM SEND-MESSAGE
*
**********************************************************************
DEFINE SUBROUTINE SEND-MESSAGE
**********************************************************************
*
  IF #I-AM-ASCII NE #CLIENT-IS-ASCII THEN
    ASSIGN SPAREPLY.TRANSLATE = TRUE
  END-IF
  ASSIGN SPAREPLY.METHOD = SPLREPLY.SEND-REPLY  /* Send with eoc
  CALLNAT 'SPUREPLY' SPAREPLY
                     SPAREPM
                     SPAETB
                     ETBCB
```

```
              MSG-INFO
END-SUBROUTINE /* SEND-MESSAGE
```

Example of Send Buffer

```
MSG1110001ATTACH-MANAGER--BOB218EC55E1AE01   AURORA-CONVERSATION FACTORY
SYSBIZ 1001 Invalid request CMD SH sent to SPSCFACT expecting 'CREATE'
```

where:

- *MSG111* is the message signature

- *0001* is the response code

- *ATTACH-MANAGER--BOB218EC55E1AE01* is the server replica ID

- *AURORA-CONVERSATION-FACTORY* is the Natural Business Services server

- *SYSBIZ* is the name of the SYSERR library used

- *1001* is the SYSERR message number

- *Invalid request CMD SH sent to SPSCFACT expecting 'CREATE'* is the message text

Business Service Interface Subprogram

Writing robust servers can be a complex task. There are many possible errors that can occur, and ensuring that each error is handled in the proper way is very difficult. Some errors are caused by resource shortages, so it is desirable to retry the call again after a brief pause. Other errors are fatal and should result in the server shutting down. Still other errors, like wait timeouts, are normal and expected.

To help simplify and standardize the task of writing servers, Natural Business Services supplies a subprogram that wraps the EntireX ACI calls. This wrapper subprogram, called SPUETB, handles many situations that have to be coded to make direct EntireX calls. To ensure that errors are handled and logged properly, use SPUETB for all EntireX calls.

Features and Benefits

This section contains a summary of the capabilities offered by SPUETB.

The following topics are covered:

- EntireX Error Handling
- Error Logging
- Shutdown Requests
- Server Timeouts

- Command Handling

EntireX Error Handling

Most EntireX errors are handled internally by SPUETB. If the errors are due to resource shortages, SPUETB pauses for two seconds and then tries the call again. The subprogram continues to retry the call for up to 20 seconds.

When implementing server receive loops, SPUETB handles all wait timeouts (EntireX error 74) and returns to the receive state.

Fatal errors cause the server to shut down if SPUETB is granted shutdown permission.

SPUETB can also handle message length errors and return a message to the sender indicating that the message was too long.

Error Logging

All errors returned from EntireX are logged in the Natural Business Services communication log. Use this log to help detect problems with your programs or environment.

Shutdown Requests

SPUETB responds to shutdown requests from EntireX. These requests can be initiated using the EntireX Control Center.

Server Timeouts

Whenever the server has not received a message for the length of time specified on the server record, the server shuts down.

Command Handling

SPUETB registers for the CMD interface and responds to all command requests. Command requests include the CMD CALLNAT command, which allows you to supply the name of the subprogram call.

SPUETB Interface

SPUETB is called using the following interface:

```
DEFINE DATA
  PARAMETER USING SPAETB    /* Specific Parameters
  PARAMETER USING ETBCB     /* Standard Broker control block
  PARAMETER
  01 SEND-BUFFER(A1/1:V)
  01 RECEIVE-BUFFER(A1/1:V)
  01 RESERVED-AREA(A1/1:V)  /* Reserved for SPUETB use
  PARAMETER USING CDPDA-M   /* Standard message area
END-DEFINE
```

As in a direct call to EntireX, the caller is responsible for filling in the EntireX control block. Additionally, the caller can specify the degree of error handling and support for common functions handled by SPUETB. This section describes these data areas.

The following topics are covered:

- SPAETB Data Area
- ETBCB Data Area
- SEND-BUFFER
- RECEIVE-BUFFER
- RESERVED-AREA
- CDPDA-M

SPAETB Data Area

The following example shows the SPAETB data area:

```
*  Data Area Name: SPAETB                    Function
*  Created on....: May 05, 98                ========
*  Created by....: SAG                This data area is passed to
*                                     SPUETB which is used to
*                                     encapsulate calls to Entire
*                                     Broker. Use SPLETB to assign
*                                     constant values.
DEFINE DATA PARAMETER
 1 SPAETB
 2 FORCE-PDA (A1/1:V) /* This field is only here
*                                /* to force the caller to create
*                                /* a separate LDA to call SPUETB
*                                /* rather than using SPAETB.
*                                /* This way, initial values can
*                                /* be placed in the LDA so that
*                                /* defaults get assigned.
 2 INPUTS
 3 METHOD (I1) /* 0 = Normal call
```

```
*                                               /* See SPLETB for other methods
 3 ENCAPSULATED-FUNCTIONS /* Set desired functions ...
 4 SUPPORT-SERVER-COMMANDS (L) /* SPUETB will automatically
*                                               /* register a command service
*                                               /* whenever a regular service is
*                                               /* registered. CMD is used as
*                                               /* the broker service name.
*                                               /* CPUETB will handle all
*                                               /* command requests directly.
 4 ALTER-RECEIVE-SERVICE (L) /* Automatically change the
*                                               /* service name on receive to
*                                               /* an '*' to allow commands
 4 SHUTDOWN-PERMISSION (L) /* If true, SPUETB is allowed to
*                                               /* shutdown the server directly.
*                                               /* See SHUT-DOWN-REASONS
 4 SHUTDOWN-REASONS /* Set desired shutdown reasons:
 5 EXPLICIT-SHUTDOWN (L) /* Shutdown request from BROKER
*                                               /* or from NBS console.
 5 TIMEOUT-REACHED (L) /* See TIMEOUT-HANDLING
 5 TERMINAL-ERROR (L) /* Non-recoverable broker error.
 4 TIMEOUT-HANDLING (I4) /* 0 = Return all timeouts so
*                                               /*     that caller can handle
*                                               /* >0= Reissue call for this
*                                               /*     many seconds. Set to
*                                               /*     max desired idle period.
*                                               /* -1= Reissue call indefinitely
*                                               /*     -1 is normally used by
*                                               /*     ATTACH servers which
*                                               /*     should run forever.
 3 ERROR-HANDLING
 4 HANDLE-TRUNCATION-ERROR (L) /* SPUETB will respond to
*                                               /* ETB error 00200094. This
*                                               /* won't be sent back to caller
 4 ERROR-HANDLER (A8) /* Normally, SPUETB will handle
*                                               /* all broker errors. However,
*                                               /* if you want to handle these
*                                               /* explicitly, supply the
*                                               /* routine to be called.
 4 USE-BUSINESS-ERROR-LOG (L) /* Log all errors on the NBS
*                                               /* file. Warning, this will
*                                               /* cause an ET to be issued.
 4 WRITE-ERRORS-TO-CONSOLE (L) /* CALL 'CMWTO' with errors
 4 WRITE-ERRORS-TO-PRINT-FILE-0 (L) /* Write errors to NATURAL
*                                               /* print file 0
 4 MAX-RETRY-TIME (I2) /* Number of seconds to continue
*                                               /* to retry call in the event of
*                                               /* a Broker resource shortage.
*                                               /* Defaults to 20 seconds.
 4 MESSAGE-DATA /* These fields are used to
*                                               /* build helpful error messages
*                                               /* when broker calls fail.
 5 CALLING-PROGRAM (A8) /* Name of caller.
```

```
 5 BUSINESS-SERVICE (A32) /* Name of business service
*                                    /* if known.
 5 CALL-DESC (A32) /* Description of the call
 2 INPUT-OUTPUTS
 3 REPLICA-ID (A32) /* Assigned at first LOGON
*                                    /* do not adjust
 3 CLIENT-MODE (L) /* In this mode, errors need
*                                    /* not be logged and checks
*                                    /* for broker error cycles
*                                    /* are not performed.
 3 OPTION (A50) /* SPUETB option
 2 OUTPUTS
 3 RESULT (I1) /* See SPLETB
*                                    /* 0 = Normal request
*                                    /* 1 = Attach request
*                                    /* 2 = Command request
*                                    /* 3 = Timeout
*                                    /* 4 = Non-terminal error
*                                    /* 5 = Terminal error
*                                    /* 6 = Restarting after error.
*                                    /*    only set after method 6
END-DEFINE
```

The fields in the SPAETB data area include:

Field	Description
FORCE-PDA (A1/1:V)	Due to the number of input settings that must be assigned before calling SPUETB, the preferred method of assigning them is to use supplied LDAs, initialized with common defaults settings. The following LDAs are supplied: ■ SPAETBC (used by client programs) ■ SPAETBS (used by server programs)
METHOD (I1)	Determines the type of processing performed by SPUETB. Assign the method values using one of the constants in SPLETB.METHODS.
SPLETB.NORMAL-CALL	Used for all EntireX calls except LOGON, REGISTER, DEREGISTER, and LOGOFF.
SPLETB.LOGON	Uses the value of SPAETB.BUSINESS-SERVICE to look up the EntireX ID, user ID, and corresponding password with which to log onto EntireX. It also executes the EntireX Logon function.
SPLETB.REGISTER-SERVER	Uses SPAETB.BUSINESS-SERVICE to look up the Broker ID, Server class, Server name, and Broker service name and uses these values to Register with EntireX. If the SUPPORT-SERVER-COMMANDS parameter is set to true, this method also registers an additional interface, CMD, to accept commands.

Field	Description
SPLETB.SHUTDOWN-SERVER	Invokes the EntireX Deregister and Logoff functions. It is only used by servers. Always issue a shutdown request before ending server programs. If you want SPUETB to perform a shut down automatically, assign `SHUTDOWN-PERMISSION=TRUE`.
SPLETB.LOG-SUPPLIED-ERROR	Requests that an application error be logged by SPUETB. The error must be passed in the MSG-INFO.##MSG field of the CDPDA-M data area. The message is logged to locations specified in the SPAETB.ERROR-HANDLING structure.
SPLETB.LOG-NATURAL-ERROR	Tells SPUETB to log the last Natural error that occurred. It is only called from ON ERROR blocks or error transactions (assign to *ERROR-TA). The error is logged to locations specified in the SPAETB.ERROR-HANDLING structure.
SPLETB.GET-SERVICE-DEFAULTS	Assigns the following fields based on the values established at the time of the initial LOGON method: ■ ETBCB.BROKER-ID ■ ETBCB.SERVER-CLASS ■ ETBCB.SERVER-NAME ■ ETBCB.SERVICE ■ ETBCB.USER-ID ■ ETBCB.TOKEN ■ ETBCB.SECURITY-TOKEN ■ SPAETB.TIMEOUT-HANDLING ■ SPAETB.BUSINESS-SERVICE
SPLETB.LOGOFF	Performs an EntireX Logoff function. For other methods, refer to SPLETB.
SUPPORT-SERVER-COMMANDS (L)	Tells SPUETB to automatically support command interface services such as PING, SHUTDOWN, etc. SPUETB automatically registers a separate interface service using CMD as the Broker service name. All command requests are handled by SPUETB; the caller need not code any specific support for commands.
ALTER-RECEIVE-SERVICE (L)	Used in conjunction with SUPPORT-SERVER-COMMANDS. If this field is set to true, SPUETB automatically changes the Broker service name specified in any Receive function to an asterisk (*). This allows the Receive function to be satisfied by either a request for the main interface service or a request for the command interface service.
SHUTDOWN-PERMISSION (L)	If true, SPUETB can shut down the current program. Normally, it is only set for server programs. To determine which events allow SPUETB to shut down the running server, assign the fields in SPAETB.SHUTDOWN. SPUETB always logs any errors prior to shutting down.

Field	Description
EXPLICIT-SHUTDOWN (L)	Allows SPUETB to shut down the server as a result of an explicit SHUTDOWN command.
TIMEOUT-REACHED (L)	Allows SPUETB to shut down the server when the server timeout value is reached. This timeout value is passed in the TIMEOUT-HANDLING parameter and defaulted from the Server Timeout field on the server record.
TERMINAL-ERROR (L)	Allows SPUETB to shut down the server in response to a fatal EntireX error.

TIMEOUT-HANDLING (I4)	Tells SPUETB how to handle timeouts when executing EntireX Receive functions. Can be one of the following:
	
	■ -1
	
	  Execute forever (use SPLETB.NO-TIME-LIMIT) to assign this value.
	
	■ 0
	
	  Return to the caller after the first receive timeout.
	
	■ >0
	
	  Execute for this many seconds, then either return to the caller or execute shutdown processing (based on SHUTDOWN-PERMISSION and TIMEOUT-REACHED parameters).
	
	This field is derived from the Server Timeout value on the server record. If no server timeout is specified, the following defaults are used:
	
	■ Servers without attach servers -1
	
	■ Servers with attach servers 1200 (= 20 minutes)

RESERVED (A8)	Reserved for future use.
USE-BUSINESS-ERROR-LOG (L)	Logs all errors to the Natural Business Services log file.
WRITE-ERRORS-TO-CONSOLE (L)	Writes all errors to the operator console.
WRITE-ERRORS-TO-PRINT-FILE 0(L)	Writes all errors to Print file 0.
MAX-RETRY-TIME (I2)	Indicates the length of time to continue trying to execute an EntireX call in the event of a resource shortage. This defaults to 20 seconds.
CALLING-PROGRAM (A8)	Identifies the caller of SPUETB. This name is used when logging error messages.
BUSINESS-SERVICE (A32)	To use the Logon and Register methods of SPUETB, specify the name of the server in this field. Also used when writing error messages.
CALL-DESC (A32)	Free-format description of the call used when logging error messages.
REPLICA-ID(A32)	Replica ID assigned to the server (output field only).

Field	Description
CLIENT-MODE(L)	If this flag is set, SPUETB does not log errors and checks for EntireX error cycles are not performed.
RESULT (I1)	Interpreted after the call to determine the results of the call. The SPLETB data area defines the following constants to check the results: ■ NORMAL-REQUEST EntireX call completed normally. ■ ATTACH-REQUEST EntireX call resulted in an attach request. Only returned to attach servers. ■ TIMEOUT Receive timeout was reached and shutdown permission for timeouts was not granted to SPUETB. ■ NON-TERMINAL-ERROR Non-terminal EntireX error occurred. This error is automatically logged by SPUETB. See ETBCB.ERROR-CODE. ■ TERMINAL-ERROR Terminal EntireX error occurred, but shutdown permission was not granted to SPUETB. The error is automatically logged.

ETBCB Data Area

ETBCB is a standard data area representing the fields that must be passed to EntireX when using the EntireX ACI. The calling program should use ETBCB12 or ETBCB13, depending on the version of the stub in use.

SEND-BUFFER

The send buffer is used in conjunction with the EntireX Send function. The size of this buffer must be greater than or equal to the value of ETBCB.SEND-LEN.

RECEIVE-BUFFER

The Receive buffer is used in conjunction with the EntireX Receive function or blocked Send functions. The size of this buffer must be greater than or equal to the value of ETBCB.RECEIVE-LEN.

RESERVED-AREA

This pass area is reserved for future use. Define and pass the SPAETBP.NOT-USED(*) parameter in place of this parameter.

CDPDA-M

This is a standard message area. Whenever SPUETB encounters a non-recoverable error, it returns with the error text in MSG-INFO.##MSG and MSG-INFO.##RETURN-CODE is assigned "E".

Use the SPUETB Subprogram

For an example of using SPUETB, refer to the SPSTIMS Timestamp Server example. If you need to do your own character set translation (because your messages contain a mixture of printable and binary data), refer to SPSTIMS2.

CMD TRACE

The TRACE command enables and disables tracing of a running server. This feature is used in conjunction with the CSUDEBI utility. The TRACE command accepts an RID to target the command to a specific replica. There are two separate forms of the TRACE command; the one you choose depends on whether you want to enable or disable tracing.

▶ **To enable tracing:**

■   Use the CMD TRACE LOCATION=n [options] command.

▶ **To disable tracing:**

■   Use the CMD TRACE OFF command.

Valid Keywords

Valid trace locations are defined in the CSLDEBUG local data area in SYSCST. The following table shows the trace keywords:

Keyword	Description
QHANDLE	A valid queue handle is required when setting the message location to 10. This is a quoted value consisting of: `'bkrid, user-ID, token, (unpacked) security-token, conv-ID'`
ERROR-TRIGGER	Forces a runtime error at a specified point within the running server. Errors can only be triggered on lines that are currently being traced. The syntax of the value assigned to this field is: `Program,Line,NATnnnn,Skip'` where: ■ *Program* is the name of the program where the runtime error is to be triggered ■ *Line* is the line number where the error is to be triggered ■ NAT*nnnn* is the error to be triggered ■ *Skip* is used if the error is not to be triggered on the next execution of the statement, but rather after executing the statement this many times
FILTER-MASK	A100 string of 0 and 1 values. "1" represents statements that are to be traced. Each mask character is related to a constant in the SPLTRACE local data area.
FILTER-PROGRAM	List of up to five programs (in quotes and separated by commas) used to limit the programs that produce trace output. You can use special characters in the program name to serve as pattern-matching characters. For details, refer to the PATTERN option for the Natural EXAMINE statement.

Example of Enabling Tracing

```
CMD TRACE
RID=BBCB0B5A1BD5AF9F201FACB0B5A14D5AF9F201,LOCATION=10,QHANDLE='BKR045,SPSCFACT,AAC
B5AB5A9201FACB5A14D5A9201,00000000000000000000000000000000000000000000000000000000,00000000000220',ERROR-TRIGGER='SPLIB,540,NAT082',FILTER-MASK='001100000001000
00010000000110000000000000000000000000000000000000,FILTER-PROGRAM='SPU*,SP?SEC'
```

Trace Response

The trace response is normally a confirmation message indicating whether the trace request was successful. The response uses the SPUREPLY protocol (MSG111).

Test the Trace Utility

To test the trace functions, use the CMD CALLNAT SPUTRTST command.

CMD CALLNAT

It is possible to CALLNAT any subprogram, provided the subprogram implements a generic interface. This interface is defined as follows:

```
DEFINE DATA
  PARAMETER USING SPACALLN  /* Standard callnat parameters
  PARAMETER USING SPAREPLY  /* Reply message parameters
  PARAMETER
  01 RECEIVE-SEND-BUFFER(A1/1:15000)
END-DEFINE
```

The CALLNAT command takes the form:

```
CMD CALLNAT subpname parameter_string
```

where:

- *subpname* is the name of the subprogram you want to CALLNAT

- *parameter_string* is any set of characters to be passed to the specified subprogram using the RECEIVE-SEND-BUFFER

For an example of how to write a new CALLNAT interface subprogram, refer to the SPUCMDT subprogram.

Conversation Factory Utility

Natural Business Services includes a facility called a Conversation Factory. This facility works in conjunction with high-level callnat and message queue APIs to facilitate the simple transfer of data between two platforms. The benefits offered by the Conversation Factory and supporting APIs include:

- Allow communication between a client and server without knowledge of EntireX ACI.

- Allow a conversation to be established between two processes, each acting as clients.

■ Support multiple concurrent conversations between the same two participants. For example, the Construct generate server listens for specifications on one conversation and cancels requests on another.

■ Are used in conjunction with servers launched from the client to establish a conversation between the client who launched a server and the server itself.

On the server, the Conversation Factory consists of the following four subprograms:

Subprogram	Description
SQUOPEN	Opens a new conversation.
SQUSEND	Sends information from one end of the conversation to the other.
SQURECV	Receives information.
SQUCLOSE	Closes the conversation.

For an example of how to use the Conversation Factory APIs, refer to the SQEXAMPL subprogram.

Multi-Tasking Verification Utility

Use this utility to verify that ADALNK has been configured to be re-entrant and that the Natural batch nucleus that uses it is also re-entrant. A re-entrant Natural nucleus is required to run servers in a batch multi-tasking environment.

To start multiple Natural subtasks, use JCL to run the supplied Natural module, TESTTASK, in batch (as documented in Natural Business Services Installation on Mainframes). If your Natural nucleus is re-entrant, TESTTASK will successfully start Natural subtask sessions that will execute the TESTSTSK program, which will then write trace information to work file 1 showing the execution status of the subtasks. Otherwise, the job that runs TESTTASK will not end and will have to be manually cancelled.

Log Utilities

Natural Business Services supplies several utilities for archiving and deleting log data. Most of the parameters apply to all log archive utilities.

The following log utilities are supplied with Natural Business Services:

- Business Service Log Utilities
- Control Record Log Utilities
- Domain Log Utilities
- Group Log Utilities

- Business Service Definition Log Utilities
- Steplib Log Utilities
- User and Group Log Utilities

Business Service Log Utilities

Utility	Description
BSBLARCP	Allows the log data for business services to be archived to a work file and optionally deleted from the log files based on a date. It also generates a log record of the archive process. This utility has the following input fields: ■ Input End Date Indicates the last LOG date to be archived. ■ Full Report Indicates which details to display. To display full details of all data being logged, enter "F" (full). To only show the main log information, enter "B" (brief). ■ Delete After Archive Indicates whether to delete log records after they are archived.
BSBLRESP	Restores data to the business service log file. It uses the entire log data created by the BSBLARCP utility. It also generates a log record of the restore process. This utility has the following input field: ■ Full Report Indicates which details to display. To display full details of all data being logged, enter "F" (full). To only show the main log information, enter "B" (brief).

Control Record Log Utilities

Utility	Description
BSCTARCP	Allows the log data for the Natural Business Services control record to be archived to a work file and optionally deleted from the log files based on a date. It also generates a log record of the archive process. This utility has the following input fields: ■ Input End Date Indicates the last LOG date to be archived. ■ Full Report

Utility	Description
	Indicates which details to display. To display full details of all data being logged, enter "F" (full). To only show the main log information, enter "B" (brief). ■ Delete After Archive Indicates whether to delete log records after they are archived.
BSCTRESP	Restores data to the control record log file. It uses the entire log data created by the BSCTARCP utility. It also generates a log record of the restore process. This utility has the following input field: ■ Full Report Indicates which details to display. To display full details of all data being logged, enter "F" (full). To only show the main log information, enter "B" (brief).

Domain Log Utilities

Utility	Description
BSDOARCP	Allows the log data for Natural Business Services domains to be archived to a work file and optionally deleted from the log files based on a date. It also generates a log record of the archive process. This utility has the following input fields: ■ Input End Date Indicates the last LOG date to be archived. ■ Full Report Indicates which details to display. To display full details of all data being logged, enter "F" (full). To only show the main log information, enter "B" (brief). ■ Delete After Archive Indicates whether to delete log records after they are archived.
BSDORESP	Restores data to the domain log file. It uses the entire log data created by the BSDOARCP utility. It also generates a log record of the restore process. This utility has the following input field: ■ Full Report Indicates which details to display. To display full details of all data being logged, enter "F" (full). To only show the main log information, enter "B" (brief).

Group Log Utilities

Utility	Description
BSGRARCP	Allows the log data for Natural Business Services groups to be archived to a work file and optionally deleted from the log files based on a date. It also generates a log record of the archive process. This utility has the following input fields: ■ Input End Date Indicates the last LOG date to be archived. ■ Full Report Indicates which details to display. To display full details of all data being logged, enter "F" (full). To only show the main log information, enter "B" (brief). ■ Delete After Archive Indicates whether to delete log records after they are archived.
BSGRRESP	Restores data to the group log file. It uses the entire log data created by the BSGRARCP utility. It also generates a log record of the restore process. This utility has the following input field: ■ Full Report Indicates which details to display. To display full details of all data being logged, enter "F" (full). To only show the main log information, enter "B" (brief).

Business Service Definition Log Utilities

Utility	Description
BSIFARCP	Allows the log data for business service definitions to be archived to a work file and optionally deleted from the log files based on a date. It also generates a log record of the archive process. This utility has the following input fields: ■ Report Type Indicates which report type to display. To include information related to the interface and method data, enter "F" (full). To only display the log for the business service header information, enter "B" (brief). ■ Delete After Archive Indicates whether to delete log records after they are archived.

Utility	Description
BSIFRESP	Restores data to the business service definition log file. It uses the entire log data created by the BSIFARCP utility. It also generates a log record of the restore process. This utility has the following input field: ■ Report Type Indicates which report type to display. To include information related to the interface and method data, enter "F" (full). To only display the log for the business service header information, enter "B" (brief).

Steplib Log Utilities

Utility	Description

BSSDARCP	Allows the log data for EntireX step library chains to be archived to a work file and optionally deleted from the log files based on a date. It also generates a log record of the archive process.
	
	This utility has the following input fields:
	
	■ Input End Date
	
	  Indicates the last LOG date to be archived.
	■ Full Report
	
	  Indicates which details to display. To display full details of all data being logged, enter "F" (full). To only show the main log information, enter "B" (brief).
	■ Delete After Archive
	
	  Indicates whether to delete log records after they are archived.

BSSDRESP	Restores data to the EntireX steplib log file. It uses the entire log data created by the BSSDARCP utility. It also generates a log record of the restore process. This utility has the following input field: ■ Full Report Indicates which details to display. To display full details of all data being logged, enter "F" (full). To only show the main log information, enter "B" (brief).

User and Group Log Utilities

Utility	Description
BSUSARCP	Allows the log data for Natural Business Services users and groups to be archived to a work file and optionally deleted from the log files based on a date. It also generates a log record of the archive process. This utility has the following input fields: ■ Report Type Indicates which report type to display. To include information related to the interface and method data, enter "F" (full). To only display the log for the business service header information, enter "B" (brief). ■ Delete After Archive Indicates whether to delete log records after they are archived.
BSUSRESP	Restores data to the users and groups log file. It uses the entire log data created by the BSUSARCP utility. It also generates a log record of the restore process. This utility has the following input field: ■ Report Type Indicates which report type to display. To include information related to the interface and method data, enter "F" (full). To only display the log for the user and group information, enter "B" (brief).