

Construct Spectrum Administration

Manual Order Number: SPE451-040IBW

This document applies to Construct Spectrum Version 4.5 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Readers' comments are welcomed. Comments may be addressed to the Documentation Department at the address on the back cover or to the following e-mail address:

Documentation@softwareag.com

© Copyright Software AG 2003

All rights reserved

Printed in the Federal Republic of Germany

The name Software AG and/or all Software AG product names are either trademarks or registered trademarks of Software AG. Other company and product names mentioned herein may be trademarks of their respective owners.

TABLE OF CONTENTS

PREFACE

Assumed Knowledge	10
Purpose and Structure of this Documentation	11
How to Use this Documentation	13
System Administration Activities	13
Application Administration Activities	14
Application Development Activities	14
Document Conventions	15
Other Resources	16
Related Documentation	16
Construct Spectrum SDK	16
Construct Spectrum	17
Natural Construct	17
Other Documentation	18
Related Courses	18

1. OVERVIEW OF THE SPECTRUM ADMINISTRATION SUBSYSTEM

Architecture of a Construct Spectrum Application	20
Mainframe Server Components	22
System Functions	24
Windows Components	25
Microsoft IIS (Internet Information Server) Components	27
Internet/Intranet Components	27
Role of the Spectrum Administration Subsystem	28
System Administration Activities	28
Application Administration Activities	28
Security Administration Activities	30
Security Cache for Construct Spectrum	30
Using the Spectrum Administration Subsystem	31
Invoking the Spectrum Administration Subsystem	31
System Administration Main Menu	32
Application Administration Main Menu	33
Using Natural Security with Construct Spectrum	34
Configuring Natural Security for Construct Spectrum	34
Starting Services in Batch Mode under Natural Security	35

Natural Security Interface to Restricted Libraries	35
System Administration Menus and Natural Security	36
Using EntireX Security with Construct Spectrum	37
Using a Hybrid Security Service	38

2. DEFINING AND MANAGING CONSTRUCT SPECTRUM SERVICES

Types of Spectrum Services	40
Main and Command Services	41
Defining Construct Spectrum Services	42
Accessing the Maintain Services Panels	42
Identifying the Spectrum Service	44
Defining Entire Broker Service Settings	44
Server Names Supplied with Construct Spectrum	45
Attach Servers	46
Linking Services to an Attach Server	46
Activation Policies.	46
Runtime Error Handling	46
Common Server Startup Interface	47
Using the START Command to Initiate Services as Subtasks	47
Using the START Command to Initiate Services Online or in Batch.	47
Defining Service Start Parameters and Routines.	48
Using Service Start Routines.	48
BATCHJOB Service Start Routine	49
Example BATCHJOB Routine	51
SUBTASKB Service Start Routine	51
Examples of SUBTASKB Routine	52
Writing to Print and Workfiles	52
Generating Trace Output	53
Defining Service Settings and Program Parameters	54
Changing the Attribute File	57
Adding a Natural Profile	57
Managing Construct Spectrum Services	58
Overview of Commands	60
DEBUG Command	61
How the DEBUG Command Works	61
ENDJOB Command	62
ENDSERVICE Command.	62
ENVIRONMENT Command	62
INITIATE Command	63
PING Command	64
REFRESH Command	64

SHUTDOWN Command	65
Shutting Down a Single Replica	65
Shutting Down All Replicas of a Service	65
Shutting Down All Servers Within a Batch Job	65
Example of Shutting Down Servers in a Batch Job	65
STEPLIBS Command	66
SYSPROF Command	67
Standard Response Protocol	67
Accessing the Communications Logs	67
Queries and Logs for Construct Spectrum Services	68
Accessing the Query Panels	68
Accessing Log Query Panels	69
3. USING THE CONSTRUCT SPECTRUM CONTROL RECORD	
Accessing the Maintain Control Record Window	72
Using the Maintain Control Record Window	73
Modifying Control Data	73
Natural Security	73
Security Exit	73
Entire Net-Work Timeout	74
Accessing the Spectrum Control Record Log	74
4. DEFINING GROUPS AND USERS	
Introduction	76
Defining Groups	78
Defining Groups Using Spectrum Security	79
Accessing the Maintain Group Table Panel	79
Defining Groups Using Natural Security	80
Accessing the Maintain Group Profile Window	80
Using the Maintain Group Profile Window	80
Defining Users	82
Defining Users Using Spectrum Security	83
Accessing the Maintain User Table Panel	83
Defining a Debug Library and File	84
Associating Users with Groups	84
Defining Users Using Natural Security	85
Accessing the Maintain User Profile Window	85
Using the Maintain User Profile Window	85
Associating Users with Groups	86
Accessing the Maintain User's Groups Window	87
Linking Users to Libraries	88
Accessing the Library Maintenance / Link Users Window	88

Queries and Logs for User and Group Tables	89
Queries and Logs for Spectrum Security.	89
Accessing Query Panels	89
Accessing Log Query Panels.	90
Queries for Natural Security	91
Accessing Query Panels	91
Query Group Table	92
Query User Table.	93
Query User's Groups.	93
5. SETTING CONSTRUCT SPECTRUM SECURITY OPTIONS	
Introduction	96
Defining Security Settings.	97
Accessing the Maintain Object Security Panels	97
Adding a Group and Domain Combination.	98
Purging a Group-Domain Combination	98
Setting Security Options	99
Setting Security Options for Group-Domain Combinations	99
Setting Security Options for Objects.	100
Setting Security Options for Methods.	101
Security Statuses	102
No Privileges Status.	102
Revoked and Granted Statuses	102
Disabled and Enabled Statuses	103
Using the Security Exit Modules.	104
BS_USEC0 (Template).	104
BS_USEC1 ("Pessimistic" Checking)	105
BS_USEC2 ("Optimistic" Checking)	105
Using the Cache Synchronization Exit Module	106
Queries and Logs for Spectrum Security.	107
Accessing Query and Log Panels	107
6. USING CONSTRUCT SPECTRUM TOOLS	
System Administration Tools	110
Accessing the System Administration Functions	110
Synchronizing Security Cache and Cleanup Queue	111
Security Cache.	111
Cleanup Queue	111
Synchronize Security Cache & Cleanup Queue Function.	111
Resetting the Security Cache.	112

Client Tools 113

 Spectrum Service Manager 113

 Service Properties Window 115

 Change Password Program 118

 Remote Clock Program 119

 Using the TIMESTAMP Service. 119

 Using the Spectrum Dispatch Client 120

 Command Line Parameters 120

 Broker Driver Program 122

7. DEPLOYING THE ADMINISTRATION SUBSYSTEM

Introduction 126

Example Deployment Scenario 127

Creating the Example Deployment Scenario 129

 Step 1: Install Software in Production Environments 129

 Step 2: Install Construct Spectrum on Each Machine. 130

 Step 3: Copy Construct Spectrum System Files 130

 Logical File 135. 130

 Logical File 136. 130

 Step 4: (Optional) Set up System File Database in Entire Net-Work 131

 Step 5: Define and Start an Attach Service 131

 Starting an Attach Service as a Started Task. 131

 Step 6: Define and Start Dispatch and Security Services 133

 Natural Security. 133

 Step 7: Transfer Domains 134

 Step 8: (Optional) Transfer Group and User Tables 134

Data Transfer Utilities 135

 Transferring Domains 135

 Accessing the Domain Transfer Utilities. 136

 Exporting Domains 137

 Importing Domains 137

 Transferring Groups 139

 Accessing the Group Transfer Utilities 139

 Exporting Groups 140

 Importing Groups 140

 Using the Data Transfer Utilities in Batch Mode 141

 Exporting Domains 141

 Importing Domains 142

 Exporting Groups 142

 Importing Groups 142

Layout of the Export Users and Groups Workfile.	143
Header Record.	143
Group Record	144
User Record.	144
APPENDIX A — PARSE USER PARAMETERS	147
BSSPARMA Parameter Data Area	147
CDPDA-M Parameter Data Area	149
APPENDIX B — ADDITIONAL ERROR-HANDLING	151
INDEX	153

PREFACE

Welcome to the Construct Spectrum Administration subsystem — your tool for setting up and managing environments for Construct Spectrum applications. *Construct Spectrum Administration* is intended for administrators who use the subsystem to manage network communications and applications. This preface will help you get the most benefit from this documentation, as well as find other sources of information about managing Construct Spectrum applications.

The following topics are covered:

- **Assumed Knowledge**, page 10
- **Purpose and Structure of this Documentation**, page 11
- **How to Use this Documentation**, page 13
- **Document Conventions**, page 15
- **Other Resources**, page 16



Assumed Knowledge

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
- Entire Broker
- Entire Net-Work

For sources of information about Natural Construct and Construct Spectrum, see **Other Resources**, page 16.

Purpose and Structure of this Documentation

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

The following table describes the information contained in each chapter:

Chapter	Title	Topics
1	Overview of the Spectrum Administration Subsystem , page 19	Introduces the architecture of Construct Spectrum client/server and web applications and the role of the Administration subsystem in supporting these applications. It also contains information on accessing and navigating the subsystem.
2	Defining and Managing Construct Spectrum Services , page 39	Describes how to create and manage Spectrum services that automate communication between application components and Entire Broker.
3	Using the Construct Spectrum Control Record , page 71	Describes how to use the Construct Spectrum control record to initiate, ping, and shutdown services, as well as how to enable Natural Security, specify a security exit, and adjust Entire Net-Work timeouts.
4	Defining Groups and Users , page 75	Describes how to define groups, users, and their associations. Groups and users are essential for creating a secure environment, whether you are using Construct Spectrum security or Natural Security.
5	Setting Construct Spectrum Security Options , page 95	Describes how to define Construct Spectrum security settings for domains, objects, and methods.
6	Using Construct Spectrum Tools , page 109	Describes the tools on the client and server for managing the environment of Construct Spectrum applications.
7	Deploying the Administration Subsystem , page 125	Describes the steps involved in deploying the Spectrum Administration subsystem to run applications in production environments.

Chapter	Title	Topics (continued)
Appendix A	Appendix A — Parse User Parameters , page 147	Describes how to use the BSSPARMN routine to parse the keywords specified in the Service Start parameter fields on the Maintain Services panel.
Appendix B	Appendix B — Additional Error-Handling , page 151	Describes how to use the SPSERRN user exit to enhance core error-processing in Construct Spectrum applications.

How to Use this Documentation

Typically, you use the Spectrum Administration subsystem for three types of activities:

- System administration
- Application administration
- Application development

The following sections describe how to use *Construct Spectrum Administration* and related Construct Spectrum documentation to perform these tasks.

Note: To learn how to access and navigate the Spectrum Administration subsystem and to gain a general understanding of the subsystem's role in supporting Construct Spectrum applications, see **Overview of the Spectrum Administration Subsystem**, page 19.

System Administration Activities

- Set up and deploy the Spectrum Administration subsystem.
 - For information about installing and setting up the Spectrum Administration subsystem, see the *Construct Spectrum and SDK Installation Guide for Windows*.
 - For information about deploying the subsystem in one or more production or development environments, see **Deploying the Administration Subsystem**, page 125.
- Set up and maintain Spectrum services.

Spectrum services interact with Entire Broker and your network to automate communication between application components on the client and server. They also handle a Spectrum application's security checking (Construct Spectrum security or Natural Security).

 - For information, see **Defining and Managing Construct Spectrum Services**, page 39.
- Access the Construct Spectrum control record to adjust Entire Net-Work timeouts, enable or disable Natural Security, and specify a security exit, if required.
 - For information, see **Using the Construct Spectrum Control Record**, page 71.
- Define users and groups to the Spectrum Administration subsystem.

The application administrator can then define the groups' access privileges to application domains, objects, and methods.

 - For information, see **Defining Groups and Users**, page 75.
- Monitor and adjust the performance of the Spectrum Administration subsystem.
 - For information about synchronizing the security cache and cleanup queue and re-setting the security cache, see **Using Construct Spectrum Tools**, page 109.
 - For information about resolving errors, see *Construct Spectrum Messages*.

Application Administration Activities

- Access the Spectrum Administration subsystem to maintain the security of applications. This involves specifying groups' access privileges to domains, objects, and methods.
 - For information, see **Setting Construct Spectrum Security Options**, page 95.
- Access the Spectrum Administration subsystem to maintain application service definitions.
 - For information, see **Using the Subprogram-Proxy Model**, page 103, *Construct Spectrum SDK Reference*.

Application Development Activities

- Access the Spectrum Administration subsystem to define a steplib chain and domain. For information, see **Setting up the Mainframe Environment**, page 37, *Construct Spectrum SDK Reference*.
- Set security options for the development environment so you can test applications.
 - For information about defining groups and users, see **Defining Groups and Users**, page 75.
 - For information about linking groups to domains, see **Setting Construct Spectrum Security Options**, page 95.
- Access the Spectrum Administration subsystem to maintain application service definitions.
 - For information, see **Using the Subprogram-Proxy Model**, page 103, *Construct Spectrum SDK Reference*.

Document Conventions

This documentation uses the following typographical conventions:

Example	Description
Introduction	Bolded text in cross references indicates chapter and section titles.
“A”	Items within quotation marks indicate values you must enter.
Browse model, GotFocus, Enter	Mixed case text indicates names of: <ul style="list-style-type: none"> • Natural Construct and Construct Spectrum editors, fields, files, functions, models, panels, windows, parameters, subsystems, variables, and dialogs • Visual Basic classes, constants, controls, windows, dialogs, events, files, menus, methods, properties, and variables • Keys
Alt+F1	A plus sign (+) between two key names indicates that you must press the keys together to invoke a function. For example, Alt+F1 means hold down the Alt key while pressing the F1 key.
CHANGE-HISTORY	Uppercase text indicates the names of Natural command keywords, command operands, data areas, help routines, libraries, members, parameters, programs, statements, subprograms, subroutines, user exits, and utilities.
<i>Construct Spectrum Administration, variable name</i>	Italicized text indicates: <ul style="list-style-type: none"> • Book titles • Placeholders for information you must supply
[<i>variable</i>]	In syntax and code examples, values within square brackets indicate optional items.
{ WHILE UNTIL }	In syntax examples, values within brace brackets indicate a choice between two or more items; each item is separated by a vertical bar ().

Other Resources

This section provides information about other resources you can use to learn more about Construct Spectrum and Natural Construct. For more information about these documents and courses, contact the nearest Software AG office or visit the website at www.softwareag.com to order documents or view course schedules and locations. You can also use the website to email questions to Customer Support.

Related Documentation

This section lists other documentation in the Construct Spectrum and Natural Construct documentation set.

Construct Spectrum SDK

- *Construct Spectrum SDK Reference*
This documentation is for developers creating Natural modules and ActiveX Business Objects to support applications that will run in the Natural mainframe environment and a Windows environment and/or an internet server.
- *Construct Spectrum SDK for Microsoft .NET Framework*
This guide is for developers creating Microsoft .NET Web services to invoke Natural subprograms (business objects) over the Inter/Intranet via the W3C SOAP standard.
- *Construct Spectrum SDK for Web Applications*
This documentation is for developers creating the web components of applications. It describes how to use the Construct Spectrum wizards in Visual Basic to generate HTML templates, page handlers, and object factory entries. It also contains detailed information about customizing, debugging, deploying, and securing web applications.
- *Construct Spectrum SDK for Client/Server Applications*
This documentation is for developers creating client components for applications that will run in a Natural mainframe (server) and Windows (client) environment.
- *Construct Spectrum Messages*
This documentation is for application developers, application administrators, and system administrators who want to investigate messages returned by Construct Spectrum runtime and SDK components.

Construct Spectrum

- *Construct Spectrum Reference*
This documentation is for application developers and administrators who need quick access to information about Construct Spectrum application programming interfaces (APIs) and utilities.
- *Construct Spectrum and SDK Vn Release Notes*
This document describes the new features, support requirements, and changes in this release of Construct Spectrum and Construct Spectrum SDK.
- *Construct Spectrum and SDK Installation Guide for Windows*
This documentation describes how to install and set up the Construct Spectrum runtime and SDK components on the client.
- *Construct Spectrum and SDK Installation Guide for Mainframes*
This documentation describes how to install and set up the Construct Spectrum runtime and SDK components on the mainframe.

Natural Construct

- *Natural Construct Installation Guide for Mainframes*
This documentation provides essential information for setting up the latest version of Natural Construct, which is needed to operate the Construct Spectrum programming environment.
- *Natural Construct Generation*
This documentation describes how to use the Natural Construct models to generate applications that will run in a mainframe environment.
- *Natural Construct Administration and Modeling*
This documentation describes how to use the Administration subsystem of Natural Construct and how to create new models.
- *Natural Construct Help Text*
This documentation describes how to create online help for applications that run on server platforms.
- *Natural Construct Getting Started Guide*
This guide introduces new users to Natural Construct and provides step-by-step instructions to create several common processes.

Other Documentation

This section lists documents published by WH&O International:

- *Natural Construct Tips & Techniques*
This book provides a reference of tips and techniques for developing and supporting Natural Construct applications.
- *Natural Construct Application Development User's Guide*
This guide describes the basics of generating Natural Construct modules using the supplied models.
- *Natural Construct Study Guide*
This guide is intended for programmers who have never used Natural Construct.

Related Courses

In addition to documentation, the following courses are available from Software AG:

- A self-study course on Natural Construct fundamentals
- An instructor-led course on building applications with Natural Construct
- An instructor-led course on modifying the existing Natural Construct models or creating your own models

OVERVIEW OF THE SPECTRUM ADMINISTRATION SUBSYSTEM

Reading this chapter will give you a general understanding of the architecture of a Construct Spectrum application and the role that the Spectrum Administration subsystem plays in supporting the application. You will also learn how to access and navigate the subsystem and how to setup security services.

The following topics are covered:

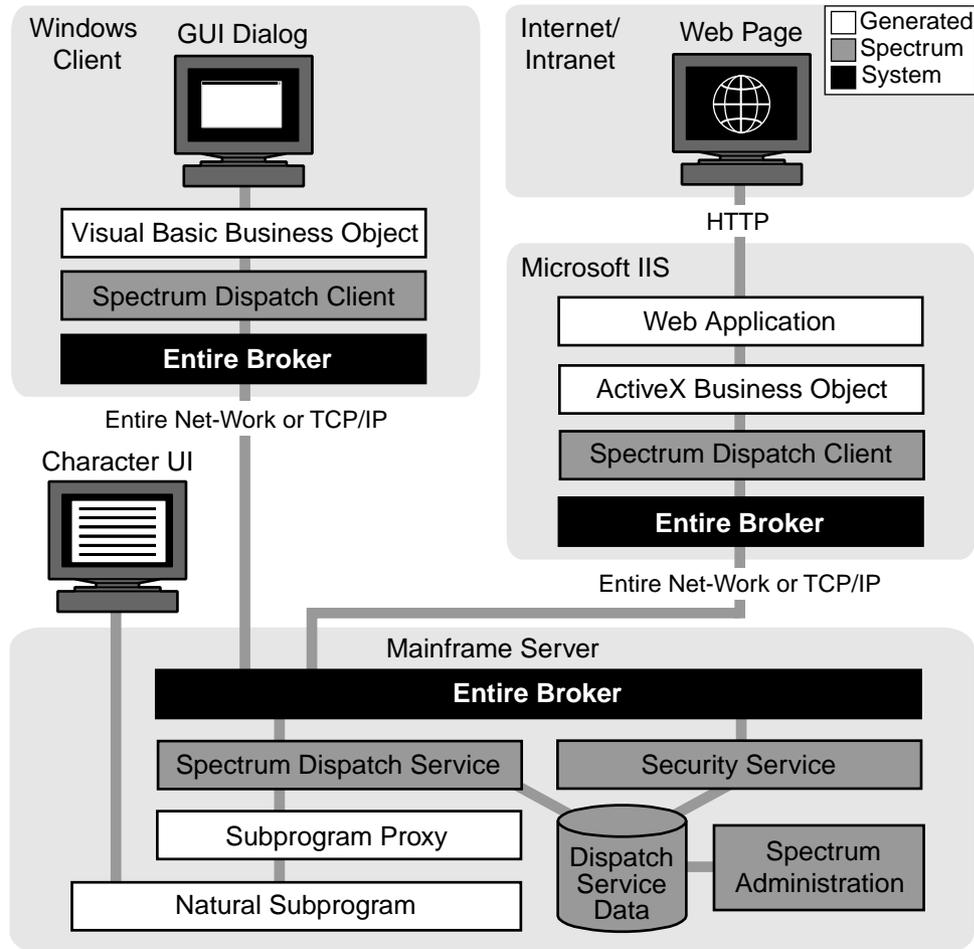
- **Architecture of a Construct Spectrum Application**, page 20
- **Role of the Spectrum Administration Subsystem**, page 28
- **Using the Spectrum Administration Subsystem**, page 31
- **Using Natural Security with Construct Spectrum**, page 34
- **Using EntireX Security with Construct Spectrum**, page 37
- **Using a Hybrid Security Service**, page 38

Architecture of a Construct Spectrum Application

Using Construct Spectrum and the software development kit (SDK), application developers can create all components of a client/server and web application, including Natural object subprograms that perform maintenance and browse functions, and GUI dialogs or web pages that communicate with the object subprograms.

Communication between server and client components of an application is performed by a combination of Entire Broker and Entire Net-Work (or Entire Broker configured to use TCP/IP), as well as Construct Spectrum's middleware components: the Spectrum Dispatch Client (SDC) and Spectrum services. The middleware components encapsulate calls to Entire Broker on the client and server.

The diagram on the following page shows the architecture of Natural character-based applications, client/server applications, and web applications.



Architecture of Construct Spectrum Applications

The following sections describe these components according to the platforms on which the components run: mainframe server, Windows, Microsoft IIS, and internet or intranet.

Mainframe Server Components

Component	Description
Natural subprograms	Perform maintenance and browse functions on the mainframe server. The same set of business objects can be accessed from character-based Natural applications, client/server applications, and web applications. This ensures that the integrity of business data is preserved, independent of the presentation layer.
Character UI	Non-distributed Natural applications created with Natural Construct accessing subprograms directly.
Subprogram proxy	<p>Acts as a bridge between a specific subprogram and the Spectrum dispatch service. The subprogram proxy:</p> <ul style="list-style-type: none"> • Provides a common interface so that the Spectrum dispatch service can pass the same set of parameters to any subprogram proxy • Issues a CALLNAT to the subprogram • Converts the parameter data of the subprogram into a format that can be transmitted between the client and server • Supports optimization of the data passed through the network so that only input parameters need to be sent to the Spectrum dispatch service and only output parameters need to be returned to the client • Validates the format and length of the data received from the client • Supports debugging features to help uncover inconsistencies between the data sent by the client and the data expected by the subprogram proxy <p>For more information, see Using the Subprogram-Proxy Model, page 103, <i>Construct Spectrum SDK Reference</i>.</p>

Component	Description (continued)
Spectrum dispatch service	<p>Provides a common interface and Entire Broker services for Natural subprograms in the application. The main functions of the Spectrum dispatch service are to:</p> <ul style="list-style-type: none"> • Receive requests from the client through Entire Broker • Optionally decompress and/or decrypt and translate the request message (see System Functions, page 24) from the client's character set (ASCII) to the server's character set (either ASCII or EBCDIC) • Check Security to ensure that the client is allowed to issue the request • Determine the name of the subprogram proxy that handles the request • Issue a CALLNAT to the subprogram proxy, passing the received message as a parameter string • Optionally compress and/or encrypt the message to be returned (see System Functions, page 24) • Send information received from the subprogram proxy back to the client application
Dispatch service data	<p>Information defined and maintained in the Spectrum Administration subsystem and accessed by Spectrum dispatch services anywhere on the network by way of Entire Broker.</p>
Spectrum administration	<p>Mainframe subsystem that allows system administrators, application administrators, and application developers to set up and manage system and application environments.</p>
Security service	<p>Checks client requests against the security settings defined in the Spectrum Administration subsystem. This stand-alone service operates independently of any one Spectrum dispatch service. Its independence allows the security service to process, in one central location, the requests of several Spectrum dispatch services, which may be located on nodes throughout the network.</p> <p>For information about security services and settings, see Setting Construct Spectrum Security Options, page 95.</p>
Entire Broker	<p>Transfers messages between Windows or the web server and the Natural environment. Entire Broker can be configured to use either native TCP/IP or Entire Net-Work as the transport layer.</p>

System Functions

All Spectrum dispatch services defined in the Spectrum Administration subsystem have access to the following common system functions:

Function	Description
Return debugging information	Ensures that all requested debugging information is generated into the source area. Debugging information is requested by setting a Trace-Option in the subprogram proxy. The debugging information is stored as a source member that can be examined or used to initiate the request locally on the server, removing the client and the network from the test.
Encrypt and decrypt data	Supplies an interface that can be called by the external (assembler or C) routines used to encrypt and decrypt data.
Compress and decompress data	Supplies an interface that can be called by the external (assembler or C) routines used to compress and decompress data.
Error handling	Manages the capturing of runtime errors, returning the errors to the client. If possible, this function also restarts the service that ended with the runtime error.
Message handling	Returns a message string based on a message number and substitution values. The function accepts and updates the data used by the Spectrum dispatch service to return the message.
Data translation	Translates data received from the client into EBCDIC or ASCII, depending on the requirements of the server.

Windows Components

Construct Spectrum client/server applications run on Windows or Windows NT.

Component	Description
Entire Broker	Transfers messages between the client and the Natural environment. Entire Broker can be configured to use either native TCP/IP or Entire Net-Work as the transport layer.
Spectrum Dispatch Client (SDC)	Component Object Model (COM) middleware component that enables Construct Spectrum applications to read from, and write to, variables in a Natural parameter data area (PDA) and to issue CALLNAT statements to Natural subprograms.

The main functions of the SDC are:

- **Natural parameter data area simulation**
The SDC defines the parameter data of Natural business objects as a series of Natural data fields, which may include structures, arrays, and redefines. To call a business object, the Construct Spectrum application must be able to assign values to these parameter data fields before calling the business object and then read the fields after the data is returned from the server.

To facilitate this, the SDC simulates Natural parameter data areas, allowing the application developer to create code that allocates a data area and reads and writes the fields in the data area. Natural parameter data areas residing in a library on the server may be downloaded (by the Construct Spectrum Add-In) to the client. This lets the SDC know the structure (field names and formats) of a parameter data area. Parameter data areas are stored in the library image file on the client and only need to be downloaded after creation or whenever they change on the server.
- **CALLNAT simulation**
The SDC allows a Construct Spectrum application to issue a CALLNAT to a Natural subprogram. All that needs to be specified in the client code is the logical name of the subprogram to be called and the list of parameter data areas to pass to the subprogram.
- **Encapsulation of Entire Broker calls**
The SDC uses Entire Broker calls to communicate with the Spectrum dispatch service. These calls are not exposed within the application layer, so the developer never needs to code Entire Broker calls.

Component	Description (continued)
	<ul style="list-style-type: none"><li data-bbox="635 331 1375 619">• Database transaction control Often, two or more calls to subprograms occur within the same database transaction such that an END TRANSACTION statement can be issued if all calls complete successfully. Also, it is advantageous to have the client application control the point at which the END TRANSACTION or BACKOUT TRANSACTION statement occurs. The SDC and the Spectrum dispatch service cooperate to provide these capabilities. <p data-bbox="635 646 1375 709">For more information, see Using the Spectrum Dispatch Client, page 193, <i>Construct Spectrum SDK Reference</i>.</p>
Visual Basic business object	Visual Basic class that acts as an intermediary between a dialog and the SDC. This class invokes the methods of subprograms on behalf of dialogs and instantiates all the data areas required to communicate with a subprogram. Visual Basic business objects can also perform local data validation to provide immediate feedback to the user without involving a network call.
GUI dialog	Graphical interface window that communicates with the user and interacts with the Visual Basic business objects and other framework components to implement business processes.

Microsoft IIS (Internet Information Server) Components

Web applications created with Construct Spectrum work with Microsoft Internet Information Server (IIS).

Component	Description
Entire Broker	Transfers messages between the web server and the Natural environment, and can be configured to use either native TCP/IP or Entire Net-Work as the transport layer.
Spectrum Dispatch Client (SDC)	Component Object Model (COM) middleware component that enables web applications to read from, and write to, variables in a Natural parameter data area (PDA) and to issue CALLNAT statements to Natural subprograms. Its main functions are simulating PDAs and CALLNATs, encapsulating Entire Broker calls, and controlling database transactions. As the client counterpart of Spectrum dispatch services, it is also responsible for such things as data marshaling, encryption, compression, error-handling, and all Entire Broker communication. For more information, see Using the Spectrum Dispatch Client , page 193, <i>Construct Spectrum SDK Reference</i> .
ActiveX business object	Encapsulates all communication with the SDC, making it efficient to invoke Natural services from the client. Each back-end business object is represented on the web server as an ActiveX object.
Web application	Consists of framework components supplied with all Construct Spectrum web projects and components you generate using the Construct Spectrum wizards (HTML templates, page handlers, and object factory entries).

Internet/Intranet Components

Construct Spectrum-generated web applications support Internet Explorer and Netscape Navigator browsers at version 4 or higher. For additional functionality:

- Internet Explorer V5 provides improved HTML rendering and the ability to bookmark web pages in Frames mode.
- Internet Explorer V5.5 and Netscape Navigator V6 support fly-out menus.

Role of the Spectrum Administration Subsystem

The Spectrum Administration subsystem consists of a table-driven server environment based in Natural. The subsystem is used for three types of activity: system administration, application administration, and security administration.

System Administration Activities

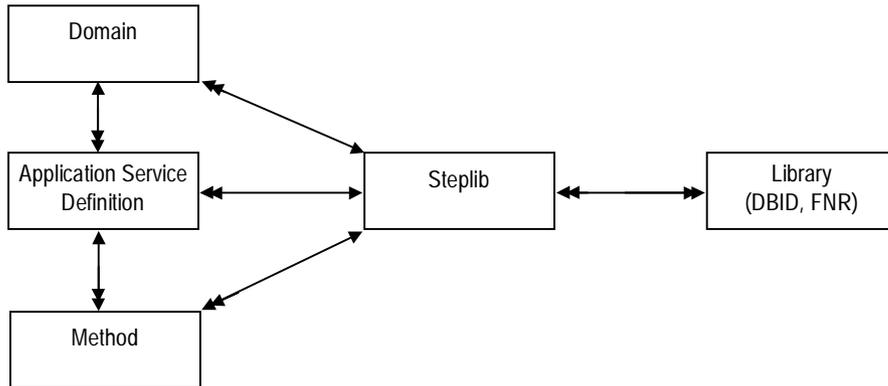
The Spectrum Administration subsystem provides dynamic, online management of tables used by Spectrum services. All data needed by services to start, stop, and perform requested functions is maintained here. The subsystem also provides mechanisms to enhance the functionality of Entire Broker, such as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

System administration involves the following tasks:

- **Defining Spectrum services**
Spectrum services, which include attach, dispatch, and security services, interact with Entire Broker and the network to automate the communication between application components on the client and server. The service definitions enable the services to make themselves known to Entire Broker and to configure themselves for the environments in which they are expected to run.
- **Managing Spectrum services**
Using the Spectrum Administration subsystem, administrators can start, stop, and check (ping) Spectrum services.
- **Viewing information logs**
Logs maintained by the Spectrum Administration subsystem contain updates to all system tables, information generated by running the Spectrum services, errors related to communication with the Spectrum dispatch service, and security violations.

Application Administration Activities

Another way of looking at the architecture of a Construct Spectrum application is to view the Spectrum Administration subsystem components that participate in directing requests from the client to the appropriate method on the server. The diagram on the following page illustrates the relationships between these components.



Application Administration Architecture

These components are:

Component	Description
Steplib and library	Defines a set of Natural libraries that a Spectrum dispatch service must access to call a subprogram proxy. The dispatch service accesses the libraries in the order they are listed in the steplib definition.
Domain	Identifier used to group business objects or define the boundaries of an application. For example, all objects in an Order Entry application (such as Customer, Order, and Product) can share a single domain.
Application service definition and method	Groups a domain, business object, and version information under one identifier that describes the business object. This definition specifies all the methods of a business object, which subprogram proxies enable access to specific methods, and where the subprograms are located (steplibs).

Application administrators and developers use the Spectrum Administration subsystem to define steplibs and domains. Application service definitions are also stored in the subsystem, where they can be modified as needed. Usually, developers create application service definitions by using a super model or the Subprogram Proxy model to generate application components.

Security Administration Activities

The Spectrum Administration subsystem is also used to define security settings that control user access to domains, business objects, 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 objects, 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 Construct Spectrum security.

Once users and groups are defined in the Spectrum 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 objects and methods in the domain.
Object	Within domains, individual objects can have security settings defined so users can only access selected objects.
Method	Access privileges can also be explicitly granted or revoked for individual methods of an object.

Security Cache for Construct Spectrum

Because security data is located in the Spectrum Administration subsystem, it can be maintained centrally. In addition, the subsystem includes a dynamically updated cache mechanism that improves throughput. When a Spectrum dispatch service receives a request from the client, the Spectrum security service assigned to it checks the user's security privileges. The Spectrum security service 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 Spectrum Administration subsystem data that affect security, the security cache is automatically updated. The security cache is maintained as a separate encrypted table on an external data source.

Using the Spectrum Administration Subsystem

This section explains how to invoke the Spectrum Administration subsystem and illustrates the menu structure. For information about navigating menus, operating panels, and using online help, direct commands, and PF-keys, see **Description of Natural Construct**, page 36, *Natural Construct Generation*.

Invoking the Spectrum Administration Subsystem

The Spectrum Administration subsystem resides in the SYSSPEC library.

➤ To invoke the Spectrum Administration subsystem:

- 1 Log onto the SYSSPEC library.
- 2 Enter “Menu” at the Next prompt.

The Construct Spectrum Administration Subsystem main menu is displayed:

```

BS__MAIN          Construct Spectrum Administration Subsystem          CDLAYMN1
Mar 11              Main Menu                                          08:55 AM

                Functions
                -----
                SA  System Administration
                AA  Application Administration

                ?  Help
                .  Terminate
                -----
Function .....  ___

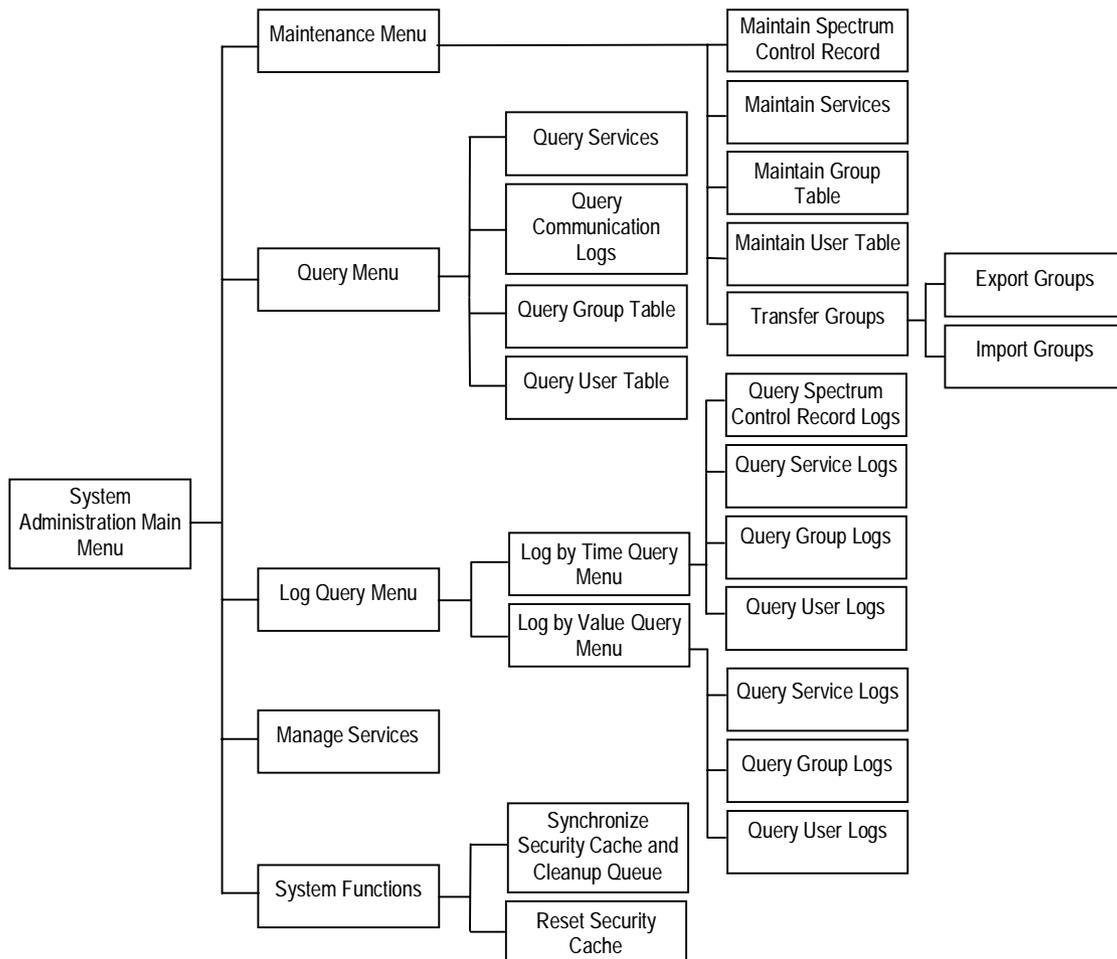
Command .....  _____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12--
      help retn quit          flip                               main
    
```

Construct Spectrum Administration Subsystem Main Menu

The Spectrum 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.

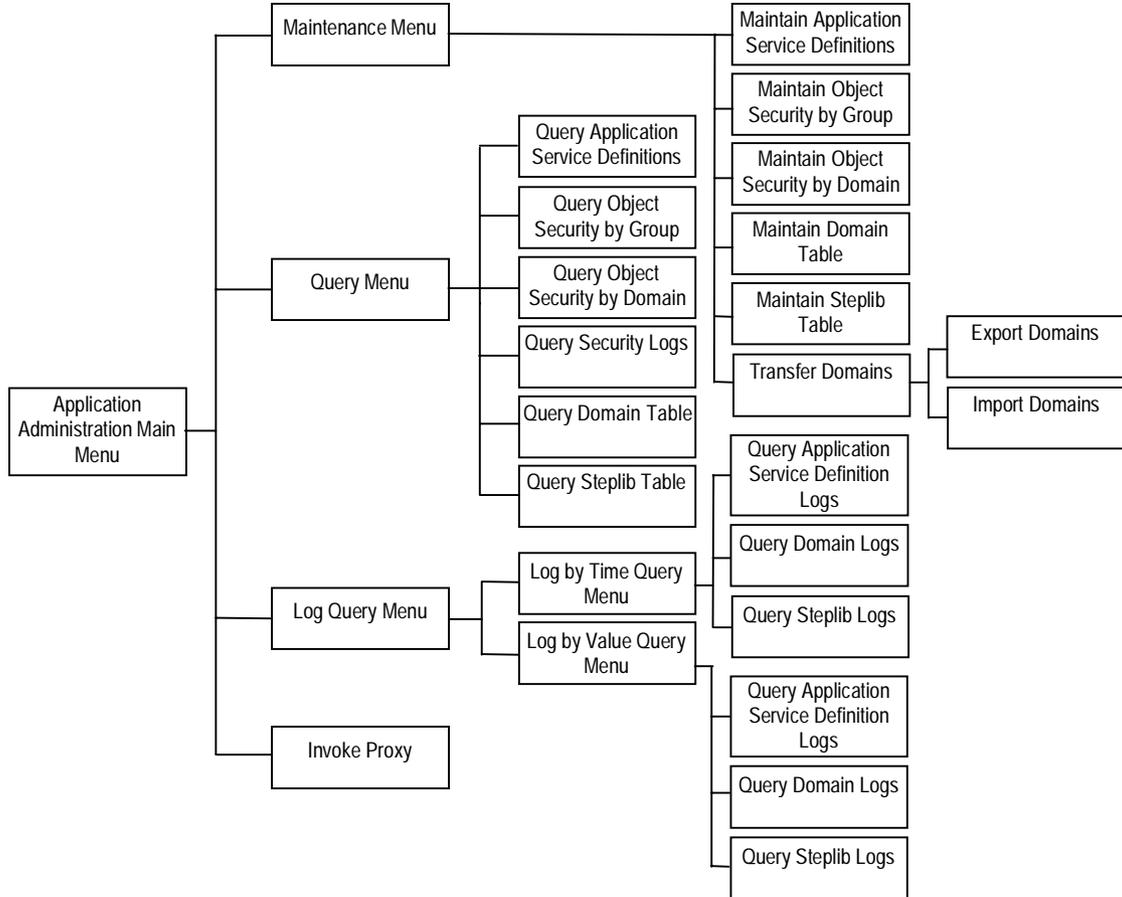
Note: If Natural Security is enabled, different options are displayed on the System Administration menus. For more information, see **Using Natural Security with Construct Spectrum**, page 34.

System Administration Main Menu



System Administration Menu Structure

Application Administration Main Menu



Application Administration Menu Structure

Using Natural Security with Construct Spectrum

The Construct Spectrum 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 Spectrum 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 Spectrum Administration subsystem or the Natural Security service.

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

For information about enabling Natural Security in the Spectrum Administration subsystem, see **Using the Construct Spectrum Control Record**, page 71.

For information about defining users and groups when Natural Security is enabled, see **Defining Groups and Users**, page 75.

Configuring Natural Security for Construct Spectrum

You can use Natural Security to maintain users and groups by configuring your Natural Security environment so that it informs Construct Spectrum whenever changes are made to user or group tables. This ensures the integrity of data in Construct Spectrum's 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 SYSSPEC 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 Construct Spectrum into your existing version of NSCUSEX1.
 - 2 Use Natural Security to add SYSSPEC 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 Spectrum (LFILE 135 and 136).

Starting Services in Batch Mode under Natural Security

When you are starting Spectrum services 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 Spectrum must be linked to the SYSSPEC library. No other special privileges need be granted to the user.

- To allow Construct Spectrum to communicate with Natural Security:
 - 1 Access 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

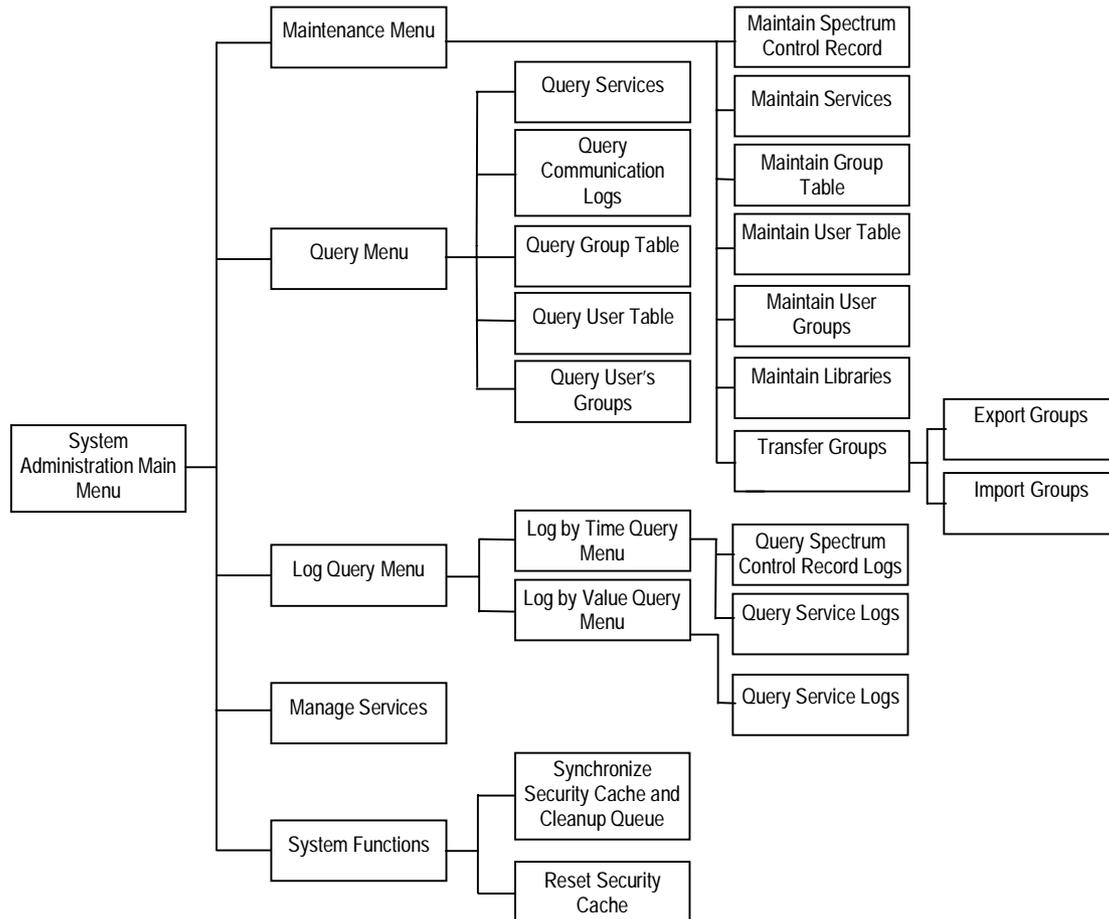
Construct Spectrum uses the published Natural Security interface, NSC---L, to determine whether a user may use a library. The interface does not provide a means to specify a DBID and FNR with the library name. If a user is linked to a library through Natural Security on a given DBID or FNR, Construct Spectrum 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 Spectrum Administration subsystem, be sure to specify only libraries that are protected by Natural Security. When you specify a steplib with a DBID or FNR in Construct Spectrum, 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 and Natural Security

When Natural Security is enabled, the options on the System Administration menus differ slightly. The following diagram shows the menu structure:



System Administration Menu Structure
Natural Security is Enabled

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 Users function invokes the Natural Security User Maintenance panel
 - Maintain Groups function invokes the Natural Security Group Maintenance panel
 - Maintain User Groups function allows you to associate users and groups
 - Maintain Libraries function allows you to define links between users and libraries

- System Administration Query menu changes include:
 - Query Users function invokes a Natural Security User Information window
 - Query Groups function invokes a Natural Security Group Information window
 - User’s Groups query function is available
 - Queries for user and group log information (by time and value) are not available

Using EntireX Security with Construct Spectrum

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

➤ To configure EntireX Security for Construct Spectrum applications:

- 1 Add a new resource class or type called “SPECTRUM”.

Note: If you call it something other than “SPECTRUM”, you must change the reference to the class or type in SPSSAF.

- 2 Create resource profiles for each domain-object-method combination used in the application using the format: `domain.object.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 Spectrum dispatch service to use the SAF security mode. For information, see **Defining Construct Spectrum Services**, page 42.
For more information about defining resources to your security service, see the EntireX Security documentation.

Note: If you modify the resource profile after the Construct Spectrum application is in use, reset the Spectrum security cache to clear previous permissions from memory. For information, see **Resetting the Security Cache**, page 112.

Using a Hybrid Security Service

Construct Spectrum also supports a hybrid security service. For example, you can use EntireX Security for user authentication (mainframe access rights) and perform validations against FSEC (Natural library and domain/object/method authorization) or Spectrum Security.

The Construct Spectrum control record definition determines which security service is used. For information, see **Accessing the Maintain Control Record Window**, page 72.

- To use a hybrid security service with a dispatch service:
- 1 Display the second Maintain Services panel for the dispatch service.
For information, see **Accessing the Maintain Services Panels**, page 42.
 - 2 Enter one of the following settings for SECURITY-MODE in the Service Start Parameters section:
 - SAF-NSC (for validations against FSEC)
The dispatcher checks application rights defined in the Natural Security file.
 - SAF-SPECT (for Spectrum Security)
The dispatcher checks application rights define in the Spectrum Security file.

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

Note: To use a hybrid security service, the Security mode setting for the dispatcher service on the client must also be set to SAF. For information, see **Service Properties Window**, page 115.

DEFINING AND MANAGING CONSTRUCT SPECTRUM SERVICES

Construct Spectrum services automate the interaction between application components and Entire Broker, freeing application developers from the task of coding calls. Spectrum services also control such application-independent facilities as security, compression, encryption, data translation, and error recovery.

Construct Spectrum provides predefined dispatch, security, and attach services that you can modify to suit your needs. You can also create your own Spectrum services and choose how services are initiated (for example, in individual batch jobs or as subtasks of an attach service).

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

The following topics are covered:

- **Types of Spectrum Services**, page 40
- **Defining Construct Spectrum Services**, page 42
- **Managing Construct Spectrum Services**, page 58
- **Queries and Logs for Construct Spectrum Services**, page 68

Types of Spectrum Services

The three types of Spectrum services are:

Service Type	Description
Attach	<p>Service that starts other services.</p> <p>When defining dispatch, security, or user-defined services, you can associate them with an attach service. If a server is intended to execute as a subtask, the attach server must be executing before the services associated with it can be started. Attach services can be used to automatically start other services on demand.</p> <p>For more information, see Attach Servers, page 46.</p>
Server	<p>Program that registers a service with Entire Broker and enters a receive loop waiting for clients to request its services. Designating a program as a server causes the common start facility (SPSSTART) to automatically logon and register with Entire Broker. Servers can also act as clients.</p> <p>Servers implement two kinds of services: main and command. For more information, see Main and Command Services, page 41.</p>
Client	<p>User of services. Client services do not register services with Entire Broker but request the use of services by sending a message to a server. When clients are started using the common start facility (SPSSTART), no implicit commands are executed. Client services are normally used only to define a Broker ID, user ID, and password required to logon to Entire Broker.</p>

Main and Command Services

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

	Server Class	Server Name	Service
Main service	SPECTRUM	DISPATCH	MAIN
Command service	SPECTRUM	DISPATCH	CMD

The main service implements commands that are specific to the server. The command 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 services allows you to automate the server startup process in response to some 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 services are:

Command	Description
PING	Determines whether a service is available.
ENVIRONMENT	Retrieves information about the environment in which the service is executing.
TRACE	Sets trace options to help debug a service.
CALLNAT	Executes a user-supplied subprogram.
SHUTDOWN	Ends a service.
ERROR	Triggers a runtime error in the server.
STEPLIBS	Lists the steplib associated with the service.
SYSPROF	Shows current LFILE settings.

Note: When sending command services programmatically, use the syntax: `CMD command`, where `command` is one of the commands listed in this table.

Defining Construct Spectrum Services

Using the Maintain Services panels in the Spectrum Administration subsystem, you can create three types of services:

- attach
- server
- client

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

Accessing the Maintain Services Panels

- To access the Maintain Services panels:
- 1 Enter “SA” in the Function field on the Construct Spectrum Administration Subsystem main menu.
The System Administration main menu is displayed.
 - 2 Enter “MM” in the Function field on the System Administration main menu.
The System Administration Maintenance menu is displayed.
 - 3 Enter “SE” in the Function field on the System Administration Maintenance menu.
The first Maintain Services panel is displayed:

```

SPBSMD      ***** Construct Spectrum Administration Subsystem *****      SPBSMD11
Aug 24      - Maintain Services -                                           1 more >

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

Entire Broker Service Settings
Broker ID.....: _____
Server class.....: _____
Server name.....: _____
Service.....: _____
User ID.....: _____
Password.....: _____
Attach Service name.....: _____ * Attach...: _

Service Start Parameters
>
>
>
Service 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

```

Maintain Services Panel 1

Use this panel to identify the service, specify Entire Broker service settings, and define service start parameters. Not all of these settings are appropriate to all types of services:

- Entire Broker service settings configure the Spectrum service to register itself by a specific name with a specific broker. You must set settings for services of type A (Attach) and S (Server). Note that services that are linked to an attach service must specify the same Broker ID as the attach service. Therefore, when the Broker ID of an attach service is modified, the Broker ID of all related services is also automatically updated.
- Most services are assigned a start routine to initiate the service in an individual batch job, or through an attach service. Two start routines are supplied with Construct Spectrum. For information about using and creating start routines for the different types of services, see **Using Service Start Routines**, page 48.

Use the second Maintain Services panel to define settings that internally configure the Spectrum service. These internal configurations include the service timeout value, default language, and program parameters. You must define some or all of these settings depending on the Spectrum service type.

```

SPBSMD      ***** Construct Spectrum Administration Subsystem *****      SPBSMD21
< 1 more          - Maintain Services -          4:58 PM

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

  Spectrum Service Settings
  Service type.....: _ (Server, Attach server, Client)

  Default language.....: __

  Service 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
Scrolling performed.
    
```

Maintain Services Panel 2

Identifying the Spectrum Service

Use the following fields on the first Maintain Services panel to identify and describe the Spectrum service:

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

Defining Entire Broker Service Settings

Use the following fields on the first Maintain Services panel to specify the parameters the Spectrum service needs to register with Entire Broker:

Field	Description
Broker ID	<p>Broker identification number. Each installation of Entire Broker is assigned a Broker ID. This number uniquely identifies the Entire Broker to your network. If you do not know the Broker ID, see your organization's network administrator.</p> <p>Specify the ID number for the Entire Broker being used. For example, if you are using an Entire Broker 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 contains only one such numeric component.</p> <p>Note: If a type S (server) service 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.</p>
Server class	Identifies the environment where the Spectrum servers are running. For example, SPECTRUM-PROD for Production services or SPECTRUM-TEST for services running in a test environment.
Server name	<p>Identifies the program that offers services. Rather than using the program name, use the server name to define a logical name for the server. This allows you to change the program name without affecting the client programs that use the service.</p> <p>For information, see Server Names Supplied with Construct Spectrum, page 45.</p>

Field	Description (continued)
Service	<p>Always assigned the value, MAIN. For information, see Main and Command Services, page 41.</p> <p>The server class/server name/MAIN service and CMD service must be defined in the Entire Broker attribute file for the specified broker. For information, see Changing the Attribute File, page 57.</p>
User ID and Password	<p>Values corresponding to values the server will use to assign the corresponding fields in the Entire Broker control block when making calls using the Entire Broker ACI (Advanced Communication Interface).</p>
Attach service name	<p>Name of the attach service used for type S (server) services.</p> <ul style="list-style-type: none"> • If the service is initiated automatically on demand, specify the attach service used to launch the service and mark the Attach field. • If the service is initiated manually, supply the attach service name; do not mark the Attach field. <p>For more information, see Attach Servers, page 46.</p>
Attach	<p>Name of the attach service used to launch the service.</p>

On startup, a service uses these parameters to tell Entire Broker that it is running and ready to handle requests. Later, you will add some of this information to the Entire Broker attribute file. (For information, see **Changing the Attribute File**, page 57.)

Server Names Supplied with Construct Spectrum

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

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

Attach Servers

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

Linking Services to an Attach Server

An attach service must be defined on the Spectrum Maintain Services panels using Service type A (attach). All services that the attach server automatically launches must specify the name of the attach service and must have the Attach field marked. This tells the attach server to register the current main service with Entire Broker using the Attach option.

For servers that are only to be launched manually, specify the attach server name without marking the Attach field. The combination of Server class, Server name, and Service must be unique for all services that are linked to a specific attach server with the Attach field marked.

An attach server can launch services as separate batch jobs or as subtasks of the attach server. For information, see **Using Service Start Routines**, page 48.

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

Activation Policies

Each time Entire Broker calls an attach server to satisfy an attach request, the attach server launches one instance of the requested service. This server continues to run indefinitely as long as it is satisfying service requests. If the server stays idle for longer than its timeout period (specified on the second Maintain Services 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 interface for starting Spectrum services is the `START` command. This section describes the different parameters you can use with the `START` command, depending on the type of service and the method of initiating it.

For Services of type `S` or `A`, the `START` command takes care of logging on to Entire Broker and registering the required Entire Broker Services. It then invokes the Natural program specified on panel 2.

Using the `START` Command to Initiate Services as Subtasks

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

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

Example of Using the `START` Command

Using the `START` Command to Initiate Services Online or in Batch

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

```
START DISPATCHER
```

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

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

Defining Service Start Parameters and Routines

The service start parameters and start routine fields are located at the bottom of the first Maintain Services panel. The service start parameters differ depending on the service start routine you use: BATCHJOB or SUBTASKB. For information, see **Using Service Start Routines**, page 48.

Although the service 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:

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

Example of Service Start Parameters

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.

Using Service Start Routines

All servers must be assigned a start routine (specified in the Service start routine field on the first Maintain Services panel). The start routine defines how the Natural session that will host the server is initiated. Two start routines are supplied with Construct Spectrum 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 service.
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 Service Start Routine

This service start routine launches a separate Natural batch job to host the server. The service start parameters must specify the name of the Natural source member that contains the JCL to be submitted. This source member is identified by the JCL keyword. All attach servers are started using the BATCHJOB start routine.

Note: To reduce the number of JCL members required, the JCL can contain place holders that can be substituted during the job submission process.

The service start keywords valid for the BATCHJOB service start routine are:

Keyword	Description
JCL	<p>Name of the Natural source member containing the JCL to be submitted. The member must exist in the library where the attach server that launches the service is running (usually the SYSSPEC library).</p> <p>The following JCL members are provided as templates in the SYSSPEC library. You can copy and modify these members to create a customized BATCHTXT member:</p> <ul style="list-style-type: none"> • BATCHJCL (OS/390) • BATCHDCL (VSE/ESA) • BATCHBCL (BS/2000) <p>For more information, see <i>Construct Spectrum and SDK Installation Guide for Mainframes</i>.</p> <p>If you place &keyword place holders in the specified JCL text member, you can use the following keywords:</p>
JOB-PREFIX	<p>If the JOBNAME is to be uniquely generated, a two-character JOB-PREFIX keyword can be specified for each service to help associate the job name with the Spectrum service. This value defaults to 'SP'.</p>
JOB-NAME	<p>To explicitly name the job, use the JOB-NAME keyword. In the JCL template, the &JOB-PREFIX&JOB-NR keyword is used to represent the JOB-NAME.</p>
	<p>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.</p>

Keyword	Description (continued)
TRANSACTION	Name of the Natural nucleus under which the service runs. This nucleus must be linked to Spectrum's 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 SPIDATA. For information, see Step 3: Load Default System File Data , page 40, <i>Construct Spectrum and SDK Installation Guide for Mainframes</i> .
NATPARM	Natural parameters used when initiating the service. The default NATPARM is: 'IM=D, PROFILE=SYSSPEC'.
LIBRARY	Name of the library to which you want to logon. This value defaults to SYSSPEC.
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 service settings, use *USER-ID.
PASSWORD	Natural Security password associated with USER-ID. If this password matches the one specified in the service settings, use *PASSWORD (which prevents the password from being displayed on the current screen).
PROGPARM	If the invoked server program accepts additional parameter values, they can be substituted for the PROGPARM keyword. These parameters can also be resubstituted with parameter values passed into SPCATT, which is used to launch a remote service from a client.

Example BATCHJOB Routine

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
```

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 Entire Broker service settings:

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

The next 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 in quotes. For example:

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

SUBTASKB Service Start Routine

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

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

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 SYSSPEC library and the required stack program. To launch the current server only, 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 service start routine supports the following keywords:

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

Examples of SUBTASKB Routine

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

SUBTASKB Parameters Without Natural Security

```
NATPARM= 'FNAT=( 1020 , 1000 ) , PROFILE=SYSSPEC ,
STACK=( LOGON SYSSPEC ; START ) '
```

Because it contains special characters, the NATPARM parameter must be in 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 Services panel is invoked.

SUBTASKB Parameters Under Natural Security

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 either:

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

```
NATPARM= 'FNAT=( 1 , 2 ) , PROFILE=SYSSPEC , AUTO=OFF ,
STACK=( LOGON SYSSPEC , *USER-ID , *PASSWORD ; START ) '
```

Writing to Print and Workfiles

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

Generating Trace Output

If you experience problems initiating or running Spectrum services, you can trace the flow of the Construct Spectrum 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 service.

Note: If you intend to use the Trace function, you must install Construct Spectrum with printer 2 and 3 assigned to batch. For information, see *Construct Spectrum and SDK Installation Guide for Mainframes*.

The following example shows the parameters to enable tracing:

```

SPBSMD      **** Construct Spectrum Administration Subsystem ****      SPBSMD11
Sep 16      - Maintain Services -                                     1 more >

Action (A,B,C,D,M,N,P)  _ Name: DISPATCH_____
                        Desc: Spectrum Dispatch Server_____

Entire Broker Service Settings
Broker ID.....: BKRL23_____
Server class.....: SPECTRUM_____
Server name.....: DISPATCH_____
Service.....: MAIN
User ID.....: SPSDISP_
Password.....:
Attach Service name.....: ATTACH_____ * Attach...: X

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

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

Example of Using the TRACE Command

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 service start routines that are applicable to other environments. Use the BATCHJOB and SUBTASKB programs as templates.

Defining Service Settings and Program Parameters

Use the second Maintain Services panel to define the Spectrum service settings used internally to configure the service:

```

SPBSMD      ***** Construct Spectrum Administration Subsystem *****      SPBSMD21
< 1 more          - Maintain Services -                                     4:58 PM

Action (A,B,C,D,M,N,P)  _ Name: _____
                        Desc: _____
    Spectrum Service Settings
    Service type.....: _ (Server, Attach server, Client)

    Default language.....: __

    Service 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
Scrolling performed.

```

Maintain Services Panel 2

The fields on this panel are:

Field	Description
Service type	<p>Type of service. Specify one of the following:</p> <ul style="list-style-type: none"> • A (attach server) • S (normal server) • C (client service) <p>The START program automatically performs a broker LOGON and Register for type A or S services.</p>
Default language	<p>Value assigned to the Natural *Language system variable on server startup.</p> <p>The service 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 service always returns to the default value after servicing a client.</p>

Field	Description (continued)
Service timeout	<p data-bbox="635 342 1362 405">Amount of time the server will continue to execute after the last time it serviced a client request.</p> <p data-bbox="635 422 1362 516">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.</p> <p data-bbox="635 533 1362 690">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.</p> <p data-bbox="635 707 1362 831">The service 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:</p> <ul data-bbox="635 842 1058 947" style="list-style-type: none">• S (seconds, for example, 3600S)• M (minutes, for example, 60M)• H (hours, for example, 3H)
Unique ID	<p data-bbox="635 974 1362 1098">16-character unique ID. This derived value is used as the prefix for the TOKEN field in the BROKER control block. This allows Spectrum to associate a running service with a Spectrum service record.</p>

Field	Description (continued)
Program Parameters	Additional parameters to allow servers to configure themselves. The following program parameters are required by the supplied servers:
Dispatch service	<p data-bbox="606 462 1375 493"><code>security-SERVER=service-name</code></p> <p data-bbox="606 504 1375 577">Name of the service that the dispatch service communicates with to retrieve security information.</p> <p data-bbox="606 588 1375 619"><code>security-MODE=value</code></p> <p data-bbox="606 630 1375 819">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 service uses the security mode to initiate the security check. This parameter can have the following values:</p> <ul data-bbox="606 829 1375 1092" style="list-style-type: none"> <li data-bbox="606 829 1375 892">• NONE No security, program names and libraries are checked <li data-bbox="606 903 1375 966">• SPECTRUM Spectrum authenticates users/checks authorizations <li data-bbox="606 976 1375 1039">• NSC Natural Security authenticates users/checks authorizations <li data-bbox="606 1050 1375 1092">• SAF SAF authenticates users/checks authorizations <p data-bbox="606 1102 1375 1176">By default, <code>security-MODE=NONE</code>. If you change the <code>security-MODE</code> value, clear the security Cache record.</p>
Attach client	<p data-bbox="606 1186 1375 1218"><code>CONVERSATION-FACTORY=service</code></p> <p data-bbox="606 1228 1375 1333">This parameter identifies the service record that defines the broker settings used to generate conversations when using the SQU* API.</p>
Natural program name	<p data-bbox="606 1344 1375 1442">When the START command is used, this parameter identifies the program invoked by the common start facility to implement the service.</p>

Changing the Attribute File

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

Note: Each Entire Broker has its own attribute file, so be sure to add the service definition to the appropriate attribute file.

You do not have to change the attribute file until you start the service. Before doing that, someone with access authority must update the Entire Broker attribute file.

For more information about the attribute file, see the Entire Broker documentation.

Adding 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. (For information about this utility, see the Natural Utilities documentation.)

In the Administration subsystem, profiles can be useful when defining Spectrum services. As part of defining a service, fill in the service start parameters. Use the NATPARM keyword to specify the system file used to store the Natural profile.

For an example of creating a Natural profile, see **Step 11: Create a New Natural Profile**, page 36, *Construct Spectrum and SDK Installation Guide for Mainframes*.

Managing Construct Spectrum Services

Use the Manage Services panel to initiate attach servers. Since all other services 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 Services 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 Services panel to shut down servers.

- To access the Manage Services panel:
- 1 Enter “SA” in the Function field on the Construct Spectrum Administration Subsystem main menu.
The System Administration main menu is displayed.
 - 2 Enter “MS” in the Function field on the System Administration main menu.
The Manage Services panel is displayed:

```

SPCMS          ***** Construct Spectrum Administration Subsystem *****          SPCMS0
Sep 17          - Manage Services -          8:15 AM

Action Spectrum Service          Type          Attach Manager
-----
___  ATTACH          Attach
___  DISPATCH        Server  ATTACH
___  SECURITY         Server  ATTACH
___  TIMESTAMP       Server  ATTACH
___  TST-ATTACH      Attach
___  TST-DISPATCH   Server  TST-ATTACH
___  TST-SECURITY    Server  TST-ATTACH
___  TST-TIMESTAMP   Server  TST-ATTACH
*** End of Data ***

Service ....: _____ Service type .: _
Replica ID .: _____ Repeat command: _
Last command:
Respondent .:

Direct Command: _____
Debug      EndJob      EndService  EnVironmnt  Initiate    Ping      Refresh
SHutdown   STeplibs   SYsprof    (PF4=CLog) (PF5=flip)

```

Manage Services Panel

Use this panel to send commands to servers and display the replies returned from the servers. This panel can also be used to initiate new servers.

By default, this panel displays type S (server) and A (attach) services. To view client services, enter “C” in the Service type field. To view all service types, enter * in the Service type field. In the Action column, enter the capitalized letter(s) of a command shown at the bottom of the panel. These commands can also be used in the Broker Driver program. For information, see **Broker Driver Program**, page 122.

The fields at the bottom of the panel are:

Field	Description
Service	Use this field to reposition the display to a specific service.
Service type	<p>Do one of the following:</p> <ul style="list-style-type: none"> • Enter “*” to show all services • Leave the field blank to show services of type A and S • Enter the code for the service type to be displayed
Replica ID	<p>ID identifying each unique server. Entire Broker 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.</p> <p>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 you finish testing, 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.</p> <p>For example, using the SHUTDOWN command for a service without specifying a replica ID shuts down any replica offering the service. 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.</p> <p>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.</p>
Repeat command	<p>Number of times to repeat the command. You can execute the command multiple times. For example, to initiate three dispatch servers, type “T” beside the dispatch service and enter “3” in this field.</p> <p>You can determine the number of servers that are running by repeating a PING command and analyzing the replica IDs of the respondents.</p> <p>Note: A better method of determining the number of servers available is by using the Entire Broker Control Center.</p>
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.

Field	Description (continued)
Respondent	Spectrum service 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 Spectrum service. If multiple services specify the same broker service, the command may be directed to an alternate Spectrum service.

You can also access the communications logs to view system messages. For information, see **Accessing the Communications Logs**, page 67.

Overview of Commands

Some commands are sent directly to the main service (MAIN) of the associated server. Other commands are sent to the command service (CMD) of 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 service of the attach server.
ENDJOB	Attach servers that launch subtasks.	MAIN service of the attach server. The attach server sends SHUTDOWN requests to linked services.
ENDSERVICE	Services of type S that are linked to an attach server.	MAIN service of the attach server. The attach service sends SHUTDOWN requests to the specified service.
ENVIRONMENT	All services of type S or A.	CMD service of the Spectrum service.
INITIATE	All service types.	If the service has an attach server, the command is sent to the attach server; otherwise, the service start routines are called directly to initiate the service.

Command	Applies to	Sent to (continued)
PING	All services of type S or A.	CMD service of the Spectrum service.
REFRESH	All services of type A.	MAIN service of the attach server.
SHUTDOWN	All services of type S or A.	CMD service of the Spectrum server.
STEPLIBS	All services of type S or A.	CMD service of the Spectrum server.
SYSPROF	All services of type S or A.	CMD service of the Spectrum server.

The Attach Service column shows the name of the attach server capable of starting the Spectrum service. Services showing highlighted attach server names are those that will automatically start in response to an Entire Broker attach request.

The following sections describe these commands in detail.

DEBUG Command

Use this command to help diagnose problems in starting services. The DEBUG command only applies to services that name 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 is sent back to the client.

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

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 service 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 shut down 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 shut down 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 services that have an attach server. In response to this command, the attach server continues to send shut down requests to the specified service until no further services 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 services that carry out the shut down 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 executing as a subtask. The PARENT-RID value identifies the attach server replica that hosts the service.

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 Entire 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 the Action field of the service.
If the system responds that the PING command was successful, there is a service of that name available on the network. If the message indicates that the service is not registered or if you require an additional copy, continue to the next step.
- 2 Enter “I” in the Action field of the service.
It can take up to 15 seconds (depending on system load) for the service to start and register with Entire Broker.

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:
 - Use the PING command to determine whether or not the service started properly.
 - If the PING command is successful, the service is now available.
 - If the PING command fails, press PF4 (CLog) to check the communications logs. These logs contain messages that are not communicated back to the client, such as startup errors. Use this information, along with information in *Construct Spectrum Messages*, to determine and rectify the problem.
 - If this is not the first copy of the Spectrum service, 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 Entire Broker Control Center to determine the number of servers available or do one of the following:
 - For attach servers or services 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 services 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 services are running and available on the network. When a service is pinged, one of the following messages is returned:

Message	Indicates
ETB Error 215/148 (NET: Connection Error)	The Broker ID specified in the service definition is not currently running or responding.
ETB Error 7/7 (Service not registered)	You have requested a valid service, 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: <i>name</i> ”, 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 service is not valid.
PING of Service <i>Service name</i> was successful	At least one copy of the service that was pinged is running and available.
ETB Error 74/74 (Wait timeout occurred)	Either all services are busy or a service has ended without properly deregistering.

REFRESH Command

The REFRESH command can only be sent to an attach server. When an attach server is started, it determines all services to be registered with the Attach option by reading the Spectrum service records. While the server is running, you can modify the service records and alter the set of services 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

Shutting Down 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.

Shutting Down All Replicas of a Service

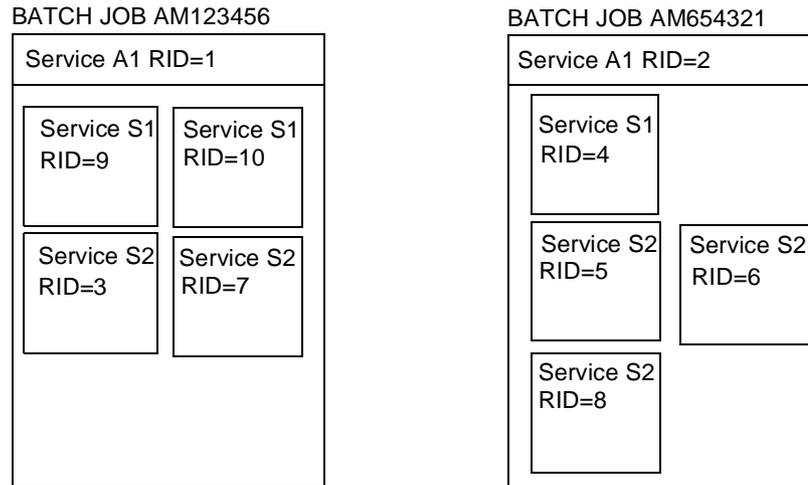
To end multiple replicas of a service, send the End Service (ES) request to the attach server used to start the service.

Shutting Down All Servers Within a Batch Job

To end all servers running within a certain batch job, issue the ENDJOB command in the Action field 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 Services panel, you can also send commands directly to servers using the Entire Broker Driver program. This section describes the commands to implement a variety of shut down options.



- To end any replica of Service S2, send the `CMD SHUTDOWN` command to the service.
- To end Service S2 Replica 5, send the `CMD SHUTDOWN,RID=5` command to the service to be shut down until you receive a successful response.
- To end all replicas of Service S2, send the `PROCESS SHUTDOWN SERVICE S2` command to either attach service.
- To end all services beginning with S2, send the `PROCESS SHUTDOWN SERVICE S2*` command to attach service A1. Note that this command will shut down Service S2 regardless of which attach server S2 is running under.
- To end all services beginning with S2 and running batch job AM654321, send the `PROCESS SHUTDOWN SUBTASK S2*,RID=2` command to the attach service. 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 service 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 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.

Accessing the Communications Logs

Access the communications logs from the Query Communications Logs panel. To access this panel, press PF4 (CLog) on the Manage Services 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 Services panel.

Queries and Logs for Construct Spectrum Services

In the Administration subsystem, you can view several queries and logs containing information about Construct Spectrum services. The following section describes the type of information each query and log shows and how to access the panels.

For information about accessing queries and logs for other Administration subsystem components, see the chapters that describe those components.

Accessing the Query Panels

- To access the query panels:
 - 1 Enter “SA” in the Function field on the Construct Spectrum Administration Subsystem main menu.
The System Administration main menu is displayed.
 - 2 Enter “QM” in the Function field on the System Administration main menu.
The System Administration Query menu is displayed:

```

BSSQMENU   ***** Construct Spectrum Administration Subsystem *****   CDLAYMN1
Aug 25     - System Administration Query Menu -                               02:13 PM

          Functions
          -----
          SE   Query Services

          CL   Query Communication Logs

          GR   Query Group Table
          US   Query User Table

          ?   Help
          .   Terminate
          -----
Function ..... ___

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

```

System Administration Query Menu

The following menu functions invoke query panels for Spectrum services:

Function	Description
Query Services	Displays existing Construct Spectrum services and their characteristics as defined on the Maintain Services panels.
Query Communications Logs	Displays messages that occurred during the operation of Construct Spectrum services.

Accessing Log Query Panels

- To access the log query panels:
 - 1 Enter “SA” in the Function field of the Construct Spectrum Administration Subsystem main menu.
The System Administration main menu is displayed.
 - 2 Enter “LM” in the Function field of the System Administration main menu.
The System Administration Log Query menu is displayed:

```

BSSLMENU          Construct Spectrum Administration Subsystem          CDLAYMN1
Apr 15             System Administration Log Query Menu              04:47 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 retn quit          flip                               main
    
```

System Administration Log Query Menu

The functions available through this menu are:

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

USING THE CONSTRUCT SPECTRUM CONTROL RECORD

The Construct Spectrum 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.

The following topics are covered:

- **Accessing the Maintain Control Record Window**, page 72
- **Using the Maintain Control Record Window**, page 73
- **Accessing the Spectrum Control Record Log**, page 74

Accessing the Maintain Control Record Window

- To access the Maintain Control Record window:
 - 1 Enter “SA” in the Function field on the Construct Spectrum Administration Subsystem main menu.
The System Administration main menu is displayed.
 - 2 Enter “MM” in the Function field on the System Administration main menu.
The System Administration Maintenance menu is displayed.
 - 3 Enter “SC” in the Function field on the System Administration Maintenance menu.
The Maintain Control Record window is displayed:

```

BSCT__MP  **** Construct Spectrum Administration Subsystem **** BSCT__11
Jan 25,02          - MAINTAIN CONTROL RECORD -                2:27 PM

Action (M)          -

Use Natural Security.....: _
security exit.....: BS_USEC2
Entire Net-Work timeout..: 50__

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

```

Maintain Control Record Window

Using the Maintain Control Record Window

Use the Maintain Control Record window to enable or disable Natural Security, specify a security exit, and set Entire Net-Work timeout values.

The following sections describe how to change these settings.

Modifying Control Data

- To modify the control data:
 - 1 Enter “M” in the Action field.
 - 2 Change the value in the appropriate field:
 - Mark the Use Natural Security field to enable Natural Security.
 - Type a new value in the security exit field to change the security exit.
 - Type a new value in the Entire Net-Work timeout field.
 - 3 Press Enter.

The following sections describe the Natural Security, security exit, and Entire Net-Work timeout values.

Natural Security

When Construct Spectrum is installed, the Natural Security field is not marked. To enable Natural Security, you must mark the field.

Note: Once Natural Security is enabled, all group and user maintenance and query functions are integrated into Natural Security panels and data. An additional query option for a user’s groups and maintenance option for User/Library link functions are added to the menu structure.

For more information about how the Construct Spectrum Administration subsystem is affected when Natural Security is enabled, see **Using Natural Security with Construct Spectrum**, page 34.

Security Exit

By default, this field displays the routine (BS_USEC2) supplied with Construct Spectrum 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.

For more information, see **Setting Security Options**, page 99.

Entire Net-Work Timeout

When Spectrum 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 the Entire Net-Work timeout field.

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

Accessing the Spectrum Control Record Log

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

- To access the Spectrum control record log:
 - 1 Enter “SA” in the Function field on the Construct Spectrum Administration Subsystem main menu.
The System Administration main menu is displayed.
 - 2 Enter “LM” in the Function field on the System Administration main menu.
The System Administration Log Query menu is displayed.
 - 3 Enter “LT” in the Function field on the System Administration Log Query menu.
The System Administration Log by Time Query menu is displayed.
 - 4 Enter “SC” in the Function field on the System Administration Log by Time Query menu.
The first Query Spectrum Control Record Logs by Time panel is displayed.

DEFINING GROUPS AND USERS

When setting up a secure environment for a Construct Spectrum application, you must create groups to represent collections of users, define users in the Construct Spectrum Administration subsystem, and associate users with the appropriate groups. This chapter describes these tasks.

The following topics are covered:

- **Introduction, page 76**
- **Defining Groups, page 78**
- **Defining Users, page 82**
- **Queries and Logs for User and Group Tables, page 89**

For related information, see:

- **Setting Construct Spectrum Security Options, page 95.**
This chapter describes the next step in setting up a secure environment — how to control a group’s access to domains, objects, and methods.
- **Creating a Domain and Setting Up Security, page 43, *Construct Spectrum SDK Reference***
This chapter contains information about creating domains.

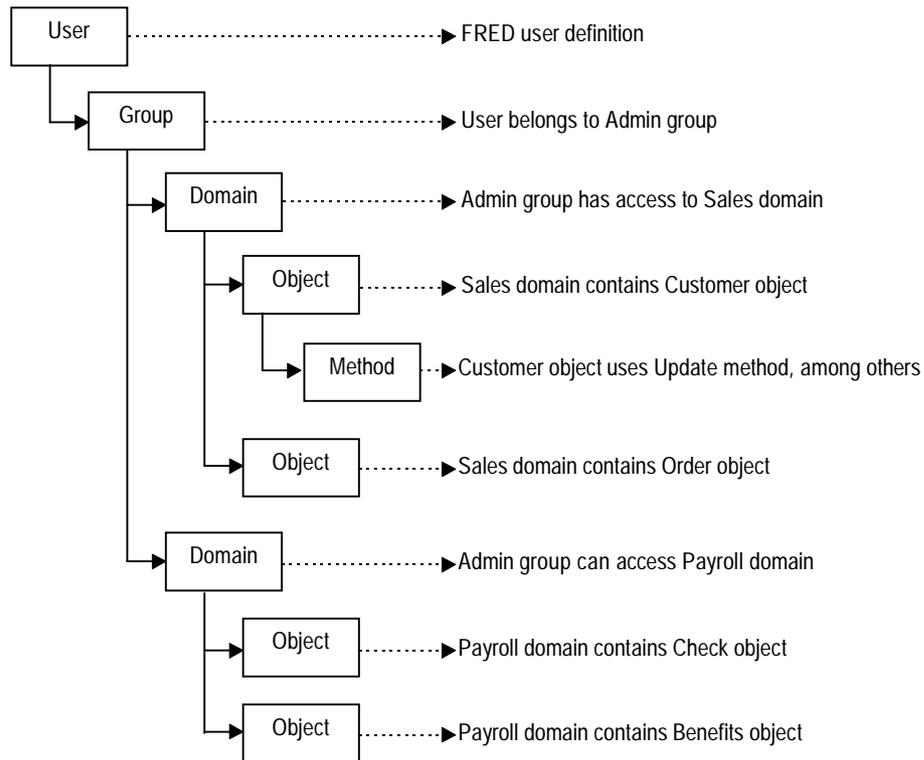
Introduction

To create a secure environment for a Spectrum application, you work with groups, users, as well as domains, objects, and object methods.

- **Groups**
You create groups to represent multiple users with the same need and authority to access particular business objects and their methods. Groups usually associate people who perform similar tasks. For example, an organization might have groups for Quality Assurance, Sales, and Management, among others.
- **Users**
You then identify individual user IDs in the Spectrum Administration subsystem. User profiles specify the groups to which they belong.
- **Domains, objects, and methods**
You create domains to collect several objects that represent business functions. In the diagram on the following page, the Sales domain encompasses the Customer and Order objects. Each object is assigned one or more methods, such as Update.

The usefulness of groups and domains becomes clear when defining security settings. With one setting, you can grant or deny a group of users access to a domain containing several objects or set security privileges for individual objects and methods. For more information, see **Setting Construct Spectrum Security Options**, page 95.

This following diagram shows the relationships between users, groups, domains, objects, and methods:



Relationship Between Users, Groups, Domains, Objects, and Methods

Because the user, FRED, is part of the Admin group, he potentially has access to the Sales and Payroll domains, their objects, and the objects' methods. You can grant or deny FRED access to an entire domain or to an individual object (such as Customer and Check) or method (such as Update).

The rest of this chapter describes how to create groups and users, using Natural Security or Construct Spectrum security only. It also describes the queries and logs available to view information about groups and users.

Defining Groups

All users of an application that employs Spectrum services must be known to the Administration subsystem through their associations with groups. To access the application's objects, a user must be a member of at least one group that is linked to a domain containing the object. 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.

Construct Spectrum allows users and groups to be defined either within the Spectrum system files or by linking to Natural Security user and group definitions. The following sections describe how to define groups both with and without Natural Security. For information, see **Using Natural Security with Construct Spectrum**, page 34.

Defining Groups Using Spectrum Security

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

Accessing the Maintain Group Table Panel

- To access the Maintain Group Table panel:
 - 1 Enter “SA” in Function field on the Construct Spectrum Administration Subsystem main menu.
The System Administration main menu is displayed.
 - 2 Enter “MM” in the Function field on the System Administration main menu.
The System Administration Maintenance menu is displayed.
 - 3 Enter “GR” in the Function field on the System Administration Maintenance menu.
The Maintain Group Table panel is displayed:

```

BSGR_MP          Construct Spectrum Administration Subsystem      BSGR_11
Feb 27           Maintain Group Table                            10:51 AM

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

```

Maintain Group Table Panel

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

Defining Groups Using Natural Security

When Natural Security is enabled in the Administration subsystem, access the Maintain Group Profile window instead of the maintenance panel supplied with Construct Spectrum.

Accessing the Maintain Group Profile Window

- To access the Maintain Group Profile window:
 - 1 Enter “SA” in the Construct Spectrum Administration Subsystem main menu. The System Administration main menu is displayed.
 - 2 Enter “MM” in the Function field on the System Administration main menu. The System Administration Maintenance menu is displayed.
 - 3 Enter “GR” in the Function field on the System Administration Maintenance menu. The Maintain Group Profile window is displayed. In the following example, the fields have been filled in for a group called “TEST”:

```

Maintain Group Profile (Natural Security)

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

Press any PF-key to stop

```

Maintain Group Profile Window

Using the Maintain Group Profile Window

- To use the Maintain Group Profile window:
 - 1 Type the code for the function you want to perform in the Function field. The options are:
 - AD (add)
 - CO (copy)
 - DI (display)
 - MO (modify)
 - 2 Type the name of the new or existing group. in the User ID field.

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

- 3 Type “G” in the User type field.
- 4 Press Enter.
The Add User panel is displayed:

```

14:26:03                *** Natural Security ***                98-03-13
                        - Add User -

User ID ..... TEST                Modified .. 96-12-04 by DEVRE
User Name .... _____
User Type .... G (Group)

Members                Libraries
-----                -----
_____                Default .. _____
_____
_____
_____                Batch User ID ..... _____

No. members 2

Additional Options ... N

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

```

Add User Panel

Note: You can use the Add User panel to add a group. A group is distinguished from an individual user by its user type: G.

For more information about using the Add User panel, see the Natural Security documentation for mainframes.

You must define all necessary group and domain combinations using the Maintain Object Security by Group panel, as described in **Adding a Group and Domain Combination**, page 98.

Defining Users

Application administrators define users so that their passwords can be checked when they attempt to perform actions in the applications supported by the 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 the 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.

Defining Users Using Spectrum Security

Use the Maintain User Table panel in the Administration subsystem to define users.

Accessing the Maintain User Table Panel

- To access the Maintain User Table panel:
 - 1 Enter “SA” in the Function field on the Construct Spectrum Administration Subsystem main menu.
The System Administration main menu is displayed.
 - 2 Enter “MM” in the Function field on the Systems Administration main menu.
The System Administration Maintenance menu is displayed.
 - 3 Enter “US” in the Function field on the System Administration Maintenance menu.
The Maintain User Table panel is displayed:

```

BSUS_MP          Construct Spectrum Administration Subsystem      BSUS_11
Feb 27           Maintain User Table                             11:07 AM

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

User ID.....: _____
Password.....: _____

Name.....: _____

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

Preferred Language.....: __

Groups.....: _____ *
              _____
              _____
              _____

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

Maintain User Table Panel

The following sections describe the debug information and how to associate users with groups.

Defining a Debug Library and File

Use the Maintain User Table panel to specify a user's debug library and file. Using Trace-Option(1), you can store data transmitted between the client and server in the debug file. For more information, see **Debugging Your Client/Server Application**, page 161, *Construct Spectrum SDK Reference*.

The Debug Library value defaults to your user ID.

Use the Debug Filename field to specify how the debug file is named:

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 HHMMSS.T. 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.

Associating Users with Groups

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

You can specify group(s) when adding a new user or modifying the profile of an existing user.

Defining Users Using Natural Security

When Natural Security is enabled in the Administration subsystem, access the Maintain User Profile window instead of the maintenance panel supplied with Construct Spectrum. 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.

The following sections describe how to access the appropriate panels and windows, add a user, associate a user with groups, and link a user to libraries.

Accessing the Maintain User Profile Window

- To access the Maintain User Profile window:
 - 1 Enter "SA" in the Function field on the Construct Spectrum Administration Subsystem main menu.
The System Administration main menu is displayed.
 - 2 Enter "MM" in the Function field on the System Administration main menu.
The System Administration Maintenance menu is displayed.
 - 3 Enter "US" in the Function field on the System Administration Maintenance menu.
The Maintain User Profile window is displayed:

```

          Maintain User Profile (Natural Security)

Function ..... AD
User ID ..... DEVRE__
User Type ..... P
Library ID or copied from user .....

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

          Press any PF-key to stop

```

Maintain User Profile Window

Using the Maintain User Profile Window

- To use the Maintain User Profile window:
 - 1 Type the code for the function you want to perform in the Function field.
Valid codes are:
 - AD (add)
 - CO (copy)
 - DI (display)
 - MO (modify)
 - 2 Type the name of the new or existing user in the User ID field.

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

- 3 Press Enter.
The Add User panel is displayed:

```

14:29:10                *** Natural Security ***                98-03-13
                        - Add User -

User ID ..... DEVRE                Modified .. 98-02-21 by DEVRE
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 ..... _1
No. groups 3          Default .. _____                Private Library ... Y
                        Last ..... _____

Additional Options ... N

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

```

Add User Panel

For more information about this panel, see the Natural Security documentation for mainframes.

Associating Users with Groups

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

After adding a user, use the Maintain User's Groups window to associate the user with one or more groups.

Accessing the Maintain User's Groups Window

- To access the Maintain User's Groups window:
 - 1 Enter "SA" in the Function field on the Construct Spectrum Administration Subsystem main menu.
The System Administration main menu is displayed.
 - 2 Enter "MM" in the Function field on the System Administration main menu.
The System Administration Maintenance menu is displayed.
 - 3 Enter "UG" in the Function field on the System Administration Maintenance menu.
The Maintain User's Groups window is displayed:

```

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

```

Maintain User's Groups Window

- To add a user to a group:
 - 1 Type the group name in the Group ID field.
 - 2 Type "LI" in the next field.
 - 3 Press Enter.
The groups members are listed in the Members columns.
 - 4 Type "AD" in the ADd / DElete / LIst members field.
 - 5 Type the user ID you want to add in an empty Members field.
 - 6 Press Enter.
The user is added to the group.

Linking Users to Libraries

Use the library link function to associate users with protected libraries. The 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

Accessing the Library Maintenance / Link Users Window

- To access the Library Maintenance / Link Users window:
- 1 Enter “SA” in the Function field on the Construct Spectrum Administration Subsystem main menu.
The System Administration main menu is displayed.
 - 2 Enter “MM” in the Function field on the System Administration main menu.
The System Administration Maintenance menu is displayed.
 - 3 Enter “LI” in the Function field on the System Administration Maintenance menu.
The Library Maintenance / Link Users window is displayed:

```

Library Maintenance / Link Users (Natural Security)

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

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

Press any PF-key to stop
```

Library Maintenance / Link Users Window

- To link a user to a library:
- 1 Type the library name in the Library ID field.
 - 2 Type the user ID in the User ID field.
 - 3 Press Enter to create the link.

Queries and Logs for User and Group Tables

In the Administration subsystem, you can view queries and logs containing information about the user and group tables. The following sections describe the options available when using Construct Spectrum security only and those available when using Natural Security.

For information about accessing queries and logs for other Administration subsystem components, see the chapters that describe those components.

Queries and Logs for Spectrum Security

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

Accessing Query Panels

- To access the group or user table query panels:
 - 1 Enter “SA” in the Function field on the Construct Spectrum Administration Subsystem main menu.
The System Administration main menu is displayed.
 - 2 Enter “QM” in the Function field on the System Administration main menu.
The System Administration Query menu is displayed:

```

BSSQMENU  ***** Construct Spectrum Administration Subsystem *****  CDLAYMN1
Aug 25      - System Administration Query Menu -                          02:39 PM

          Functions
          -----
          SE  Query Services

          CL  Query Communication Logs

          GR  Query Group Table
          US  Query User Table

          ?  Help
          .  Terminate
          -----
Function .....  _

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

```

System Administration Query Menu

- Enter the code for the appropriate function in the Function field.
The panel for the specified function is displayed.

The following menu functions invoke query panels for group and user tables:

Function	Description
Query Group Table	Displays existing groups and their characteristics as defined on the Maintain Group Table panel.
Query User Table	Displays existing users and their characteristics as defined on the Maintain User Table panel.

Accessing Log Query Panels

- To access the log query panels for the group and user tables:
 - Enter “SA” in the Function field on the Construct Spectrum Administration Subsystem main menu.
The System Administration main menu is displayed.
 - Enter “LM” in the Function field on the System Administration main menu.
The System Administration Log Query menu is displayed:

```

BSSLMENU          Construct Spectrum Administration Subsystem          CDLAYMN1
Apr 15            System Administration Log Query Menu                 04:47 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
  
```

System Administration Log Query Menu

- Enter either “LT” (Log by Time Query) or “LV” (Log by Value Query).
The LT option presents values in time order, beginning with the most recent. The LV option presents values (group names, for example) in alphanumeric order. Both options show log information, such as actions, dates, and times.

Queries for Natural Security

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

Accessing Query Panels

To access any of the query panels:

- 1 Enter “SA” in the Function field on the Construct Spectrum Administration Subsystem main menu.
The System Administration main menu is displayed.
- 2 Enter “QM” in the Function field on the System Administration main menu.
The System Administration Query menu is displayed:

```

BSSQMENS   ***** Construct Spectrum Administration Subsystem *****   CDLAYMNI
Aug 25      - System Administration Query Menu -                           02:41 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

```

System Administration Query Menu

- 3 Enter the appropriate function in the Function field.
The Natural Security window for the specified function is displayed.

Query User Table

Enter “US” in the Function field to display the List Users window:

```

List Users (Natural Security)

User Type ..... P_ (A,B,M,P,T)
User ID ..... _____
Next start value ..... _____

-----

User ID          Name
-----

```

Press any PF-key to stop

List Users Window

Use this window to view all users of type “P” (Person). Enter a user ID or starting value in the appropriate fields to scroll to a particular user.

Query User’s Groups

Enter “UG” in the Function field to display the List Groups for a User window:

```

List Groups for a User (Natural Security)

User ID ..... _____
Start group value ..... _____

Press any PF-key to stop

```

List Groups for a User Window

Enter the user ID to view the groups to which the user belongs. You can also enter a starting value to scroll to a particular group. The Display User panel is displayed:

```
14:12:56          *** Natural Security SYSTEM ***          98-05-06
                  - Display User -
The following groups contain the user: DEVRE

No. User ID      User Name      Default  No. of  Group
-----
1 TEST   Test Group 0      Library Members  Type    No
-----
2 TEST2  Test Group 2      Library Members  Type    No
3 TEST3  Test Group 3      Library Members  Type    No

-----
Enter no. to be processed: __      Reposition to: _____      Exit:  _
-----
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10---PF11---PF12---
      Help      Exit  AddOp      Flip      Canc
```

Display User Panel

SETTING CONSTRUCT SPECTRUM SECURITY OPTIONS

This chapter describes how to set up security for applications. Security options control a user's access to the domain, objects, and methods for an application. When the client requests access to data on the server, the Spectrum dispatch service invokes the Spectrum security service, which then executes the security exit. This Natural routine analyzes the security settings to determine if the user is allowed to access the data. If the user has permission, the call proceeds. If the user is not allowed access to the data, a message is returned to the client and the call ends.

This chapter describes how to define the security settings and how to use the security exits provided with Construct Spectrum to interpret the security settings.

The following topics are covered:

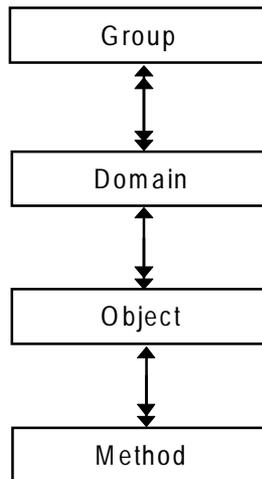
- **Introduction, page 96**
- **Defining Security Settings, page 97**
- **Using the Security Exit Modules, page 104**
- **Using the Cache Synchronization Exit Module, page 106**
- **Queries and Logs for Spectrum Security, page 107**

For related information, see:

- **Defining and Managing Construct Spectrum Services, page 39**
This chapter describes how to create a Spectrum security service.
- **Defining Groups and Users, page 75**
This chapter describes the relationships between users, groups, domains, objects, and methods. It also describes how to set up security at the system level.

Introduction

Spectrum security allows you to define a group's access privileges at multiple levels: domain, object, and method. The following diagram illustrates this hierarchy:



Levels of Security

However, you do not have to define security settings for every object and method. By default, security privileges provided at one level automatically apply to all lower levels in the hierarchy. And different settings can always be explicitly defined for individual items at a lower level. As well, Construct Spectrum 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 an application service definition is ignored in security definitions. When adding a new version of an object that is already defined to security, the new version is automatically assigned the same privileges as older versions.

Defining Security Settings

This section describes how to access the Maintain Object Security panels and how to use the panels to set access privileges at several levels.

Accessing the Maintain Object Security Panels

You can choose between two ways of viewing security information: arranged alphanumerically either by group or by domain. The following steps describe how to access either option.

- To access the Maintain Object Security panels:
 - 1 Enter “AA” in the Function field on the Construct Spectrum Administration Subsystem main menu.
The Application Administration main menu is displayed.
 - 2 Enter “MM” in the Function field on the Application Administration main menu.
The Application Administration Maintenance menu is displayed.
 - 3 Do one of the following:
 - Enter “OG” for the Maintain Object Security by Group panel, which shows groups in alphanumeric order.
 - Enter “OD” for the Maintain Object Security by Domain panel, which shows domains in alphanumeric order.

The following example shows the Maintain Object Security by Group panel:

```

BSOSG_MP          Construct Spectrum Administration Subsystem          BSOSG_11
Aug 17            Maintain Object Security by Group                    9:39 AM

Act      Group      Domain      Current      Previous
-----  -
-        SAMPLE      SAMPLE      Granted
-        SPEC01          SPEC01      Granted
-        SPEC02          SPEC02      Granted
-        SPEC03          SPEC03      Granted

Group: _____ * Domain: _____ *
Direct command...:_____
(PF4=Add)      Clear      Disable      Enable      Grant      Objects      Purge
(PF5=flip)

```

Maintain Object Security by Group Panel

Adding a Group and Domain Combination

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

➤ To add a group and domain combination:

- 1 Type the name of the group in the Group field.
- 2 Type the name of the domain in the Domain field.
- 3 Press PF4 (Add).

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

Adding a group-domain 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. The following sections describe how to access security levels and assign security statuses.

Note: If you add a new object 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 object to which it belongs.

Purging a Group-Domain Combination

The Purge action is only available at the group-domain level. It removes all security statuses for the group-domain combination and its associated objects 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 action also removes the group-domain combination.

Setting Security Options

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

For information about the security statuses you can assign to group-domain combinations, objects, and methods (such as grant or disable), see **Security Statuses**, page 102.

Setting Security Options for Group-Domain Combinations

You can define settings for any combination of group and domain. If a setting is defined only at the group-domain level, the group has the same privilege for all objects and methods within the domain. For more information, see **Adding a Group and Domain Combination**, page 98.

- To set security options for a group-domain combination:
 - Enter the appropriate code in the Act field for the group. The Maintain Object Security by Group panel is displayed:

Act	Group	Domain	Current Access	Previous Access
---	-----	-----	-----	-----
-	CST411T	CST411T	Granted	
-	CST411T	DEMO	Granted	
-	CST411T	SAGDEMO	Granted	
-	SYSTEM	BKR-COMM	Granted	
-	SYSTEM	DEMO	Granted	
-	SYSTEM	SAMPLE	Granted	

BSOSG_MP Aug 17 Construct Spectrum Administration Subsystem Maintain Object Security by Group BSOSG_11 10:21 AM

Group: _____ * Domain: _____ *

Command.....: _____

(PF4=Add) Clear Disable Enable Grant Objects Purge

(PF5=flip)

Maintain Object Security by Group Panel

Use the Clear action to remove security settings for a group-domain combination. This action also clears the group-domain's settings from the security cache.

Setting Security Options for Objects

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

➤ To set security options at the object level:

- 1 Enter “O” in the Act field for the group-domain combination.

A window is displayed showing the objects that belong to the domain. For example:

```

BSOSO_MN          Construct Spectrum Administration Subsystem
Aug 17            Security for Group CST411T/Domain DEMO: Granted  10:32 AM

Act              Object              Current Previous
-----          -----          -----
- CUSTOMER
- NO-PARMS
- ORDER                      Revoked
- PRODUCT
- SYSCST_CUSPMA
- WAREHOUSE
-
*** End of Data ***
Object:
Command.....: _____
Clear  Disable  Enable  Grant  Methods  Revoke
(PF5=flip)

```

Security for Group-Domain Window

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

- 2 Enter the appropriate code in the Act field for the object.
Use the Clear action to remove security settings for an object. This action also clears the object's settings from the security cache.

Setting Security Options for Methods

Use the third level in the security hierarchy to allow the group access to specific methods of an object or to override settings defined at the object or group-domain level.

- To set security options at the method level:
- 1 Enter “M” in the object’s Act field in the Security for Group-Domain window. The Security for Object window is displayed:

```

BSOSM_MN          Construct Spectrum Administration Subsystem
Aug 17             Security for Object CUSTOMER: None          10:33 AM

Act               Method           Current Previous
-----          -----          Access  Access
-----          -----          -----
- BROWSE
- DELETE
- EXISTS          Revoked
Method:
Command.....:
Clear           Disable      Enable      Grant       Revoke      (PF5=flip)

```

Security for Object Window

The security status for the object is displayed in the second header. The security status applied to the object is not shown for each of its methods in the Current Access field, since a status defined for an object applies to associated methods by default. Only a status defined for an individual method is displayed in the Current Access field.

- 2 Enter the appropriate code in the Act field 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

You can assign security statuses to any group-domain combination, object, or method in the security hierarchy. To apply the enable, disable, grant, or revoke statuses, enter the appropriate code in the Act field next to the group-domain combination, object, or method, as described in the previous section.

The following sections describe the generic uses of the statuses. How they are interpreted when a client request is made to the dispatch service depends on the security exit you are using. For more information, see **Using the Security Exit Modules, page 104**.

No Privileges Status

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

A blank Current Access field has a different meaning when it belongs to an object or method. When you set a group-domain combination's security status, all of its objects and their methods "inherit" that status. However, their Current Access fields remain blank unless you assign statuses to them individually. When you explicitly set an object's security status, that status applies to its methods as well.

Revoked and Granted Statuses

The granted status allows the group access to the domain, object, or method to which the status is assigned. The revoked status prevents users from accessing the group-domain combination, object, 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 group-domain combination) for each group to which a user belongs. A user can access a method that has the granted status, although the object 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 group-domain has the status, granted, and the object's status is revoked, the object and its methods are inaccessible. Even if one of the methods of the object has access explicitly granted, the revoked status at the object level disallows access to any of the methods.

Disabled and Enabled Statuses

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 group-domain combination, object, 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.

Using the Security Exit Modules

Construct Spectrum provides exit modules (BS_USEC*) you can exchange or modify to interpret the security settings differently. These exits receive information from the Spectrum security service 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.

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.

To switch security exits, access the Maintain Spectrum Control Record panel and replace the current exit. Then restart all security services. For more information, see **Using the Construct Spectrum Control Record**, page 71.

The following sections describe the security exits supplied with Construct Spectrum. BS_USEC2 is the default security exit.

BS_USEC0 (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 application level into this template.

BS_USEC1 (“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, object, 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 (“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, object, 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

```

Using 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 object 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 group-domain 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 object level, all entries for the domain-object 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 object.

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

Note: Do not specify that deletion start at a lower level than that being updated. For example, if an object'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.

For more information about security cache synchronization, see **Synchronizing Security Cache and Cleanup Queue**, page 111.

Queries and Logs for Spectrum Security

In the Administration subsystem, you can view queries and logs containing information about Spectrum security settings.

Accessing Query and Log Panels

- To access the query and log panels:
 - 1 Enter “AA” in the Function field on the Construct Spectrum Administration Subsystem main menu.
The Application Administration main menu is displayed.
 - 2 Enter “QM” in the Function field on the Application Administration main menu.
The Application Administration Query menu is displayed:

```

BSAQMENU          Construct Spectrum Administration Subsystem          CDLAYMN1
Aug 17            Application Administration Query Menu              02:38 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

```

Application Administration Query Menu

- 3 Enter the appropriate function in the Function field.
The query panel for the specified function is displayed.

The following menu functions invoke query panels for security information:

Function	Description
Query Object Security by Group	Displays groups in alphanumeric order, as well as their domains, objects, methods, and security settings.
Query Object Security by Domain	Displays domains in alphanumeric order, as well as the groups, objects, methods, and security settings.
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.

USING CONSTRUCT SPECTRUM TOOLS

Construct Spectrum provides several tools for managing applications and client/server communications. This chapter describes how to use these tools in the Administration subsystem and on the client.

The following topics are covered:

- **System Administration Tools, page 110**
- **Client Tools, page 113**

System Administration Tools

This section describes several functions in the Administration subsystem to maintain the system and system data. Using the System Administration Functions menu, you can access tools to synchronize the security cache and cleanup queue, as well as reset the security cache.

Accessing the System Administration Functions

- To access the System Administration Functions menu:
 - 1 Enter “SA” in the Function field on the Construct Spectrum Administration Subsystem main menu.
The System Administration main menu is displayed.
 - 2 Enter “SF” in the Function field on the System Administration main menu.
The System Administration Functions menu is displayed:

```

BSSAMENU   ***** Construct Spectrum Administration Subsystem *****   CDLAYMN1
Aug 25     - System Administration Functions Menu -                       03:18 PM

          Functions
          -----
          SQ   Synchronize security Cache & Cleanup Queue
          SC   Reset security Cache

          ?   Help
          .   Terminate
          -----
Function ..... _

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

```

System Administration Functions Menu

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

Synchronizing Security Cache and Cleanup Queue

This section describes the purpose of the security cache and cleanup queue and when to use the Synchronize Security Cache & Cleanup Queue function.

Security Cache

When a Spectrum dispatch service receives a request from the client, the security service assigned to it checks the user's security privileges. If lookups had to be made to a user table, then a group table, then an application 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 service 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 module 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, Construct Spectrum 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.

Synchronize Security Cache & Cleanup Queue Function

Use the Synchronize Security Cache & Cleanup Queue function to complete any cache cleanup tasks that failed prior to completion, such as when clients performing remote CALLNATs receive the "SPE5065> 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 execute this function at any time and as often as necessary.

Resetting the Security Cache

Use this function to completely clear all entries from the security cache. By emptying the cache, the security service 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 service or change the SAF resource profiles, reset the security cache to clear the old settings.

Client Tools

The following client tools are supplied with Construct Spectrum:

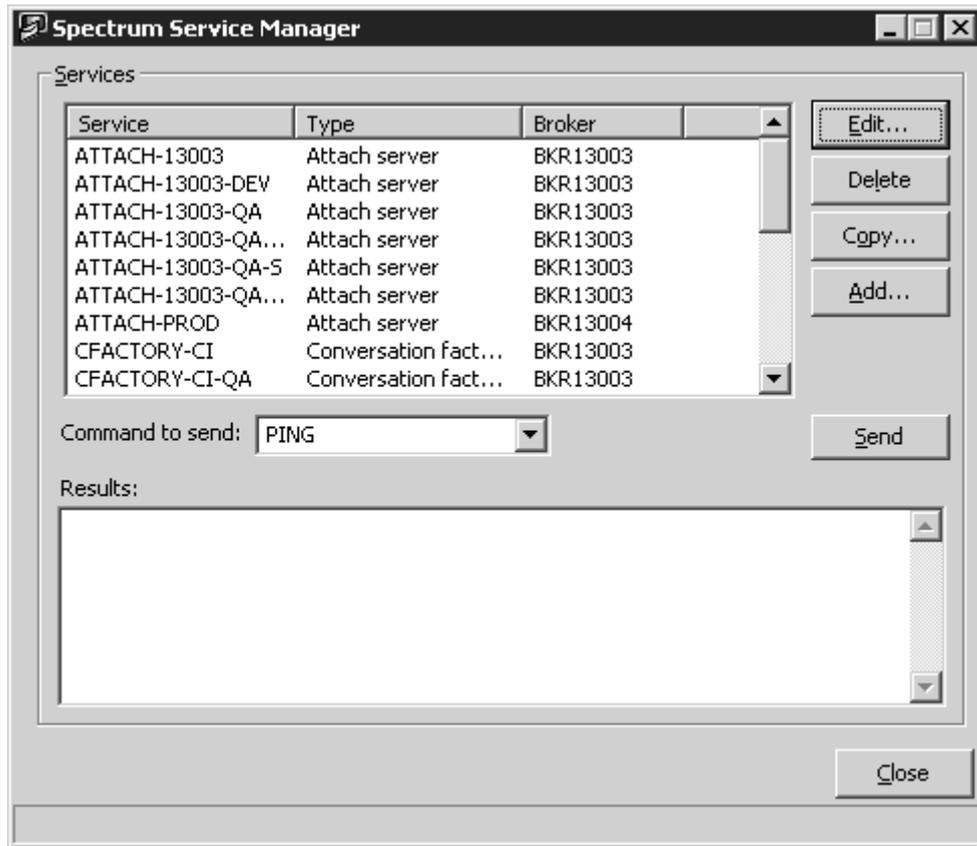
Tools	Description
Spectrum Service Manager	Configures the client to access Spectrum dispatch services running on one or more servers. Both administrators and application developers use this program.
Change Password	Allows users to change a Construct Spectrum password on the server.
Remote Clock	Displays the current time on the server. This program can be implemented in either of two ways: using the Timestamp service or using the Spectrum Dispatch Client.
Broker Driver	Allows you to interactively make calls such as Send, Receive, and Register.

Spectrum Service Manager

For the client to communicate with a Spectrum service running on a server in the network, the client needs to know how the service identifies itself to Entire Broker. This information includes the broker ID, server class, server name, and service and is stored in a special file called SDC.INI in the Windows directory on each PC. The Spectrum Service Manager allows you to edit the contents of this file. Use the Spectrum Service Manager to do the following:

- Define Spectrum services you want the client to access.
- Ping Spectrum services to determine if they are running and to test the communications path between the client and server. The ping function ensures that DLLs required by Construct Spectrum are installed and in the path. They are checked in the following order:
 - BROKERV.B.DLL (from ETB\BIN)
 - CDED32.DLL (from Windows\System)

The following example shows the Spectrum Service Manager:



Spectrum Service Manager

The Spectrum installation program always installs one Spectrum dispatch service definition, called DISPATCH.

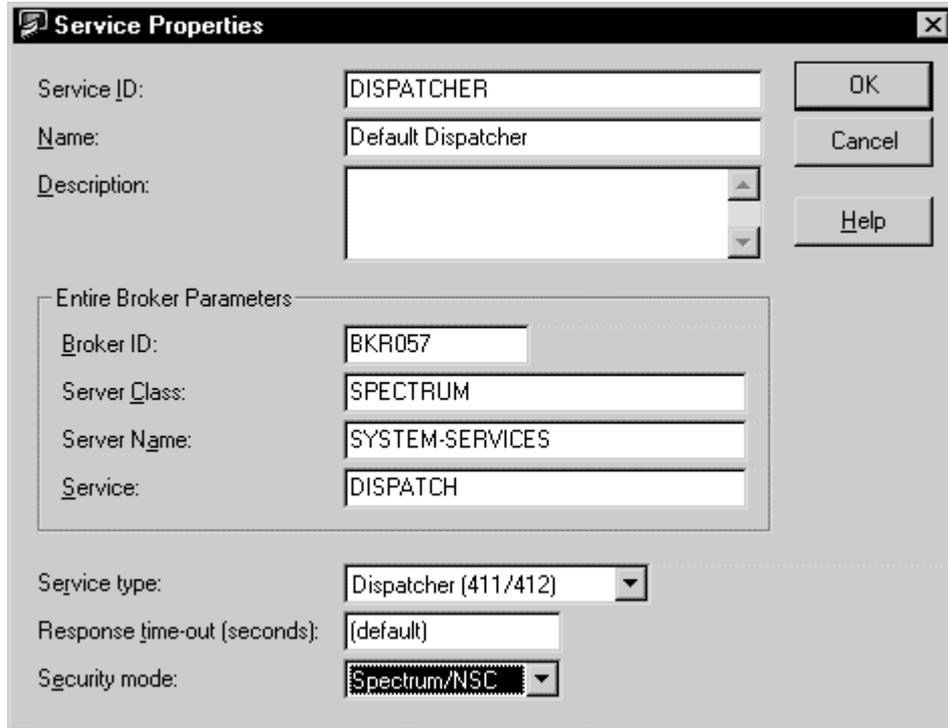
To send a message to a Spectrum service when testing the communications path between the client and server, click Send to send the PING command. If the message “PING of Service <name> was successful” is displayed, the communications path is working. Any communication problems are displayed on the status line or in a message window.

If you receive the ETB02150148 error, indicating that Entire Broker is not active, a possible cause may be that some of the DLLs required by the Entire Broker stub or the Entire Net-Work client are not in the Windows path. By default, the installation program for Entire Net-Work does not add the ADALNK and ETB\BIN directories to the Windows search path.

To resolve the error, ensure that the root directory for Entire Net-Work is in the path, as well as the ADALNK and ETB\BIN subdirectories. The Readme file for Entire Net-Work indicates that you must do this before starting Entire Net-Work.

Service Properties Window

To invoke the Service Properties window, click Edit, Copy, or Add:



The screenshot shows a dialog box titled "Service Properties". It contains the following fields and controls:

- Service ID:** DISPATCHER
- Name:** Default Dispatcher
- Description:** (empty text box)
- Entire Broker Parameters:**
 - Broker ID:** BKR057
 - Server Class:** SPECTRUM
 - Server Name:** SYSTEM-SERVICES
 - Service:** DISPATCH
- Service type:** Dispatcher (411/412)
- Response time-out (seconds):** (default)
- Security mode:** Spectrum/NSC

Buttons on the right: OK, Cancel, Help.

Service Properties Window

The Service ID text box specifies the unique name by which a service is known to Spectrum on the client. The Name text box identifies the Spectrum service in Construct Spectrum applications. To avoid confusion, we recommend that you use the same name as defined for the Spectrum service in the Administration subsystem.

If you have more than one Spectrum service, each name must uniquely identify the service and should help determine where the service is running or what its primary role is, such as Production Dispatcher or Development Dispatcher.

The Description text box value is optional. This description is used only in the Spectrum Service Manager for comments about the Spectrum service.

The Entire Broker parameters define the broker ID, server class, server name, and service that uniquely identify a Spectrum service to Entire Broker. These settings must match the settings in the Administration subsystem. For example:

```

SPBSMD      ***** Construct Spectrum Administration Subsystem *****      SPBSMD11
Aug 25      - Maintain Services -                                          1 more >

  Action (A,B,C,D,M,N,P)  _ Name: DISPATCHER_____
                          Desc: Dispatcher (Production)_____

    Entire Broker Service Settings
  Broker ID.....: BKR057_____
  Server class.....: SPECTRUM_____
  Server name.....: DISPATCH_____
  Service.....: MAIN
  User ID.....: _____
  Password.....: _____
  Attach Service name.....: ATTACH_____ * Attach...: X

    Service Start Parameters
  > ATTACH=ATTACH,TRANSACTION=NATBTCH2,
  > NATPARM='FNAT=(19,26),PROFILE=SYSSPECD'
  >
  Service start routine...: SUBTASKB *      Source S421      Object S421

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

```

Maintain Services Panel 1

The Service type text box specifies the following types of service:

Service Type	Description
Dispatcher	Spectrum dispatch service, version 4.2.1 or higher.
Dispatcher (411/412)	Spectrum dispatch service, version 4.1.1 or 4.1.2. (Older dispatch services use a different Ping protocol.)
Attach Server	Spectrum attach service.
Conversation Factory	Spectrum conversation factory.
Other	Any other type of Spectrum service that you may want to Ping, such as security servers or your own servers.

If you are viewing a dispatch service, the following information is also displayed in the Service Properties window:

- The Response time-out value specifies how long the Spectrum Dispatch Client waits for a response from the Spectrum service before returning with a “Server is not responding” message. The default value is 15 seconds. You can set it to any value from 1 to 60 seconds.
- The Security mode value specifies the security system the dispatcher is using. This setting must match the service’s SECURITY-MODE parameter as defined in the Administration subsystem:

```

SPBSMD      ***** Construct Spectrum Administration Subsystem *****      SPBSMD11
Aug 25      - Maintain Services -                                          1 more >

Action (A,B,C,D,M,N,P)  _ Name: DISPATCH_____
                        Desc: Spectrum Dispatch Server_____
Entire Broker Service Settings
Broker ID.....: BKR059_____
Server class.....: SPECTRUMDEV_____
Server name.....: DISPATCH_____
Service.....: MAIN
User ID.....: SPSDISP_
Password.....:
Attach Service name.....: ATTACH_____ * Attach..: X

Service Start Parameters
> NATPARAM='FNAT=(19,30),PROFILE=S421BTCH,
> STACK=(LOGON S421;START)'
> SECURITY-MODE=SAF
Service start routine...: SUBTASKB *      Source S421      Object S421

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

```

Maintain Services Panel 2

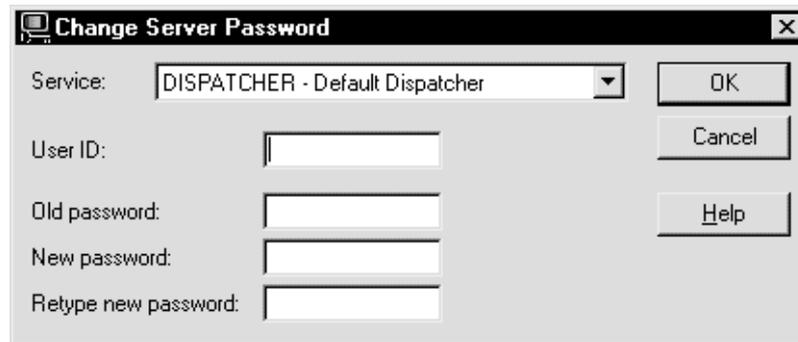
Tip: If you are distributing a Construct Spectrum application on multiple clients, you can copy the Spectrum service definitions to other PCs. The definitions are stored in the SDC.INI file in the Windows directory.

Change Password Program

Note: If you configured Spectrum to use Natural Security, you cannot use the Change Password program to change your password.

Each Construct Spectrum user (developer, administrator, user, etc.) has a user ID and password defined in the Administration subsystem. Depending on your network configuration, this user ID and password may or may not be the same as that used to access Windows, your local area network, or other server resources.

The Change Password program allows users to change their Construct Spectrum password without accessing the Administration subsystem:

The image shows a Windows-style dialog box titled "Change Server Password". It has a standard title bar with a minimize button, a maximize button, and a close button (X). The dialog contains several input fields and buttons. On the left side, there are labels for "Service:", "User ID:", "Old password:", "New password:", and "Retype new password:". The "Service:" field is a dropdown menu currently showing "DISPATCHER - Default Dispatcher". The other four fields are empty text boxes. On the right side, there are three buttons: "OK", "Cancel", and "Help".

Change Server Password

- To change your password:
- 1 Select a Spectrum service from Service.
 - 2 Type your user ID in User ID.
 - 3 Type your current password in Old password.
 - 4 Type your new password in New password and Retype new password.
 - 5 Click OK.
A confirmation message is displayed.

Remote Clock Program

The Remote Clock program displays a clock that shows the current time of day on the server. It obtains the server's current time of day using one of two methods:

- Using a user service, called the `TIMESTAMP` service, which is started from the Administration subsystem. This service returns the value of the `*TIMESTMP` Natural system variable.
- Using the Spectrum Dispatch Client to do a remote `CallNat` to a simple Natural subprogram. This subprogram also returns the value of `*TIMESTMP`.

The Remote Clock program converts the `*TIMESTMP` value into the current date and time and then formats it for display. The following example shows what the Remote Clock looks like when running:



Remote Clock

The two options are described in the following sections.

Using the `TIMESTAMP` Service

The `TIMESTAMP` service is a Natural program that illustrates the basics of writing user-defined Spectrum services that can be defined and controlled with the Administration subsystem. The Spectrum dispatch service and Spectrum security service are more complex services, but they have essentially the same general structure as the `TIMESTAMP` service.

When the `TIMESTAMP` service starts, it registers a service with Entire Broker. It then enters a receive loop waiting for requests from a client application. It responds to Ping and Shutdown messages from the Administration subsystem. When it receives a Shutdown message, it deregisters its service from Entire Broker and then ends. Much of this functionality is encapsulated in the reusable `SPUETB` routine.

The client application that communicates with the `TIMESTAMP` service is the Remote Clock program. This program and the `TIMESTAMP` service use direct calls to communicate with each other. Thus, the round-trip time from client to server and back to client is short.

Using the Spectrum Dispatch Client

The Remote Clock program can also use the Spectrum Dispatch Client to do a remote CallNat. In this case, it calls the following subprogram:

```
DEFINE DATA
  PARAMETER
    01 #TIMESTAMP(A13)
  LOCAL
    01 #LOCAL-TIMESTAMP(A16)
END-DEFINE
MOVE EDITED *TIMESTAMP(EM=H(8)) TO #LOCAL-TIMESTAMP
#TIMESTAMP := #LOCAL-TIMESTAMP
END
```

This subprogram converts the *TIMESTAMP value into a sequence of 16 hex digits, of which the first 13 are significant in Natural. These 13 hex digits are returned in the #TIMESTAMP output parameter.

When using this method to obtain the *TIMESTAMP value, Construct Spectrum takes care of all of the calls and data marshaling so that the client application and the subprogram it calls can be simple.

Command Line Parameters

The Remote Clock program accepts a series of command line parameters to control its operation. The full syntax of the command line is:

```
RemoteClock.exe [options]
```

where options can be one or more of the following:

Option	Description
-s	Use the Spectrum Dispatch Client instead of direct calls. The default is to use direct calls.
	Note: The -l, -c, -e, -d, -u, and -p options are only valid when using the -s option; otherwise they are ignored.
-l	Display performance measurements.
-c	Use compression. The default is no compression.
-e	Use encryption. The default is no encryption.
-d <i>dispatch-service</i>	Use this dispatch service definition from the Spectrum Service Manager. The default is to use the first definition in the SDC.ini file.
-t <i>title</i>	Display this title in the window title. The title cannot include spaces.

Option	Description (continued)
<i>-a time-adjustment</i>	<p>Adjusts the time returned by the server. Time-adjustment has the following format:</p> <p><i>dDhHmMsStT</i></p> <p>where <i>d</i>, <i>h</i>, <i>m</i>, <i>s</i>, and <i>t</i> specify positive or negative units of days, hours, minutes, seconds, or tenths-of-a-second, respectively. For example:</p> <pre>RemoteClock -a -2d6h</pre> <p>adjusts the time returned by subtracting two days and adding six hours.</p> <p>Use this option if the value returned by *TIMESTAMP has not been adjusted to your local time zone, for example.</p>
<i>-u user-ID</i>	Use this user ID to perform remote CallNats.
<i>-p password</i>	Use this password to perform remote CallNats.

Broker Driver Program

The Broker Driver program allows you to interactively make calls such as Send, Receive, and Register. This program is invaluable when you are developing client or server applications and want to test how a particular function works or behaves with a certain set of input parameters.

Note: The Broker Driver program sets API-TYPE to 1 and API-VERSION to 2. You cannot use it to test version 3 calls, such as unit of work.

The example on the following page shows the Broker Driver window.

Broker Driver

Broker ID: Automatic logon
User ID:
Token:
Environment: Security Token:

Function: API Version: 2 3
Service:
Conversation ID:
Send Data:
Wait: Client User ID:
Option: Conversation Status:
Receive Length: Return Length:
User Data: Convert from EBCDIC Display Hex
Receive Data:

Unit of Work

ID:	<input type="text"/>	User Data:	<input type="text"/>
Store:	<input type="text" value="0 - None"/>	AD Count:	<input type="text" value="0"/>
Time:	<input type="text"/>	Status:	<input type="text" value="0 - None"/>
Status Persist:	<input type="text"/>		

Last Broker Message:
Round-trip Time:

Broker Driver Window

If you are familiar with the ACI, you will recognize many of the fields in the Broker Driver window. Only a few text boxes require explanation:

- Use the Automatic logon check box when EntireX security is installed. When this check box is selected and a security error occurs, such as an invalid user ID or password, the Broker Driver program displays a logon window that lets you enter your user ID, password, and, optionally, your new password. If this check box is not selected, enter the user ID, password, and new password into the fields provided.
- The Security token text box shows the contents of the security-TOKEN field in the Broker control block. Because this field is binary, the contents are displayed as a series of hex pairs.
- Use the Service text box to enter the SERVER-CLASS, SERVER-NAME, and SERVICE values in the Broker control block. Enter these values in this order, with a slash character between values. For example: SPECTRUM/DISPATCH/CMD. The Service text box also contains a drop-down list and remembers all values entered into this field. Retrieve a previous value by selecting it from the drop-down list.
- Enter a conversation ID into the Conversation ID text box or choose one of the pre-defined values: NONE, NEW, ANY, or OLD.

After entering all values for a Broker call, click Go! to execute the call. After the call completes, the error class, error number, and error text are displayed at the bottom of the window, along with the total round trip time in milliseconds. If any Broker control block fields were updated by the call (such as the conversation ID), the corresponding fields in the window display those values.

DEPLOYING THE ADMINISTRATION SUBSYSTEM

This chapter describes the steps to deploy the Administration subsystem in development and production environments.

The following topics are covered:

- **Introduction**, page 126
- **Example Deployment Scenario**, page 127
- **Creating the Example Deployment Scenario**, page 129
- **Data Transfer Utilities**, page 135
- **Layout of the Export Users and Groups Workfile**, page 143

For related information, see:

- **Deploying Your Client/Server Application**, page 189, *Construct Spectrum SDK Reference*

This chapter describes how to deploy a Construct Spectrum application.

Introduction

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

- All modules related to the Administration subsystem.
- All modules related to the Construct Spectrum dispatch, security, and attach services.
- One or more copies of the system files.
- At least one dispatch, security, and attach service.

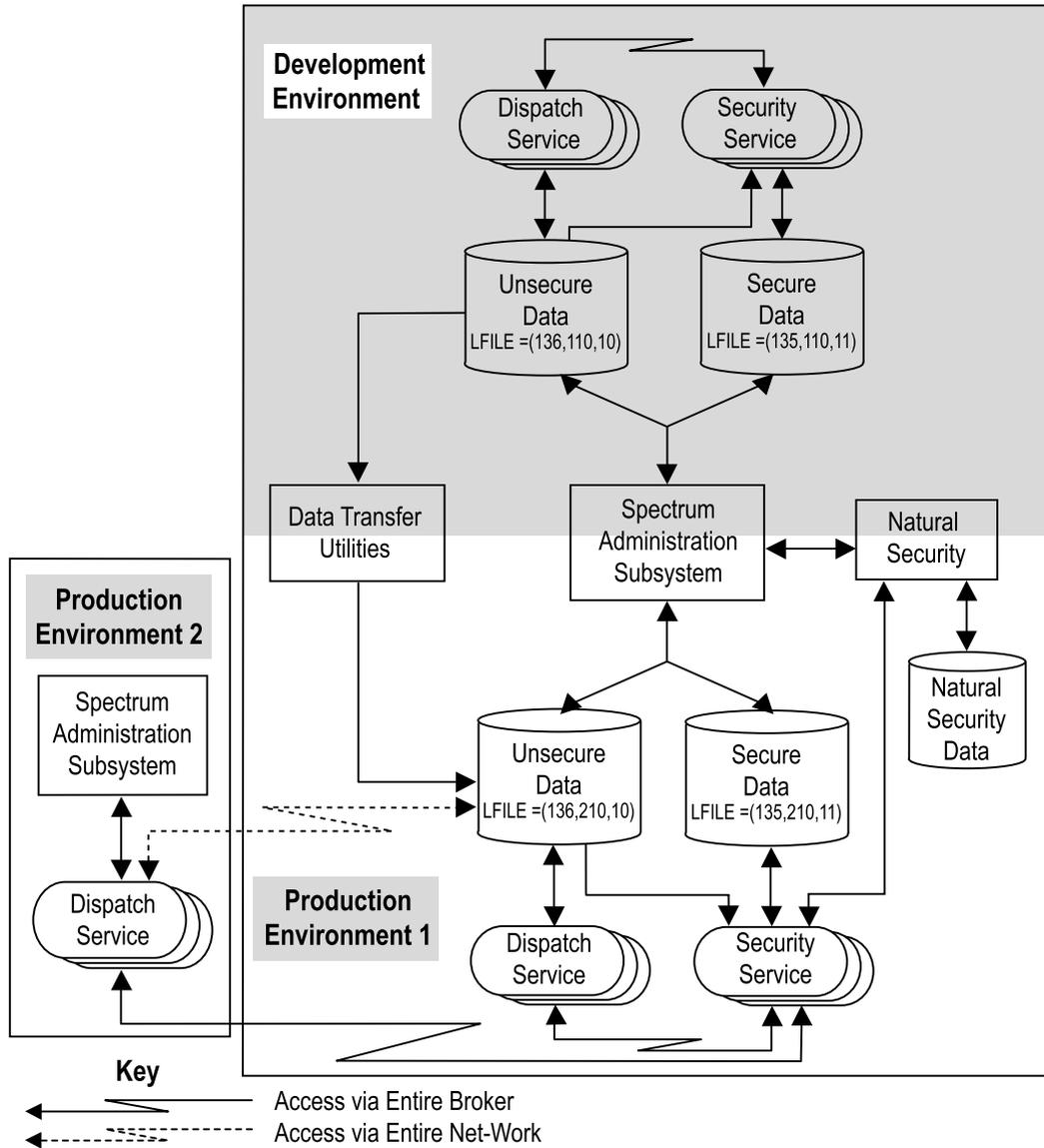
Each runtime environment accesses the same centrally-located data by means of direct calls through one of the following:

- Entire Net-Work
- Entire Broker configured to use TCP/IP as the network transport protocol
- Spectrum security service (in the case of secure data)

Example Deployment Scenario

The diagram on the following page 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 Construct Spectrum system files. In this scenario, Entire Net-Work is installed on both machines. Entire Broker 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 service is required, and both copies of the Natural Security software must be configured to update the Spectrum system files with security information.



Example Deployment Environment

Creating the Example Deployment Scenario

This section describes the prerequisites and tasks involved in setting up the sample scenario shown on page 127. Refer to these steps when planning your own deployment strategy:

- To set up the sample deployment scenario:
 - ❑ **Step 1: Install Software in Production Environments**, page 129
 - ❑ **Step 2: Install Construct Spectrum on Each Machine**, page 130
 - ❑ **Step 3: Copy Construct Spectrum System Files**, page 130
 - ❑ **Step 4: (Optional) Set up System File Database in Entire Net-Work**, page 131
 - ❑ **Step 5: Define and Start an Attach Service**, page 131
 - ❑ **Step 6: Define and Start Dispatch and Security Services**, page 133
 - ❑ **Step 7: Transfer Domains**, page 134
 - ❑ **Step 8: (Optional) Transfer Group and User Tables**, page 134

Each of these steps is described in the following sections.

Step 1: Install Software in Production Environments

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

- Entire Broker must be installed on one or more machine. In the example, 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 Spectrum application, 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.

For information about product versions, see **Prerequisites**, page 14, *Construct Spectrum and SDK Installation Guide for Mainframes*. For information about installing and configuring these products, see their respective installation guides.

Step 2: Install Construct Spectrum on Each Machine

The Administration subsystem defines and manages Spectrum services and maintains Administration subsystem data. You can access a full set of queries and log data in the subsystem.

Construct Spectrum must be installed on each computer. The sample scenario on page 127 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 about installing Construct Spectrum, see

- *Construct Spectrum and SDK Installation Guide for Mainframes*
- *Construct Spectrum and SDK Installation Guide for Windows*

Step 3: Copy Construct Spectrum System Files

In the example scenario, Production environment 1 shares the Administration subsystem with the Development environment. While it is possible to share the Spectrum 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 Spectrum dispatch service, the Spectrum security service, and the 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 dispatcher information to allow this sensitive data to be enciphered, if desired.

Logical File 136

Logical file 136 contains all information used by Construct Spectrum, except the user and security information stored on logical file 135.

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

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

Step 5: Define and Start an Attach Service

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

Starting an Attach Service as a Started Task

Typically, you start attach services in an online environment, which is resource intensive. If desired, you can start an attach service as a started task. Although you will be starting the service from the console, as opposed to from within Construct Spectrum, you can continue to use all other features in the Spectrum Administration system.

- To run an attach service 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=SPE451'
//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))
//DDSORTTUT 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=SPE451.JCL( spectrum input data)
//DDCARD DD *
ADARUN PROGRAM=USER,DATABASE=13001,MODE=MULTI,SVC=249
```

- 2 Initiate the attach service from the Spectrum Administration menu.
- 3 Enter “%%” to display the JCL submitted to the editor.
In the example, the JCL is contained in `spectrum input data`.

Step 7: Transfer Domains

Decide which applications 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 Administration subsystem. When you use the utilities to copy the table data from one environment to another, the domains, application service definitions and, optionally, the steplib chains are also copied.

For information about importing and exporting domains, see **Transferring Domains**, page 135.

Step 8: (Optional) Transfer Group and User Tables

If users and groups have already been defined in the production environments, omit this step. Otherwise, use the Export Groups and Import Groups utilities to copy this data to your production environments. For more information about exporting and importing groups, see **Transferring Groups**, page 139.

If you use the Export and Import Groups utilities, the permissions that grant access between groups and domains are not transferred. You must set security definitions in the Administration subsystem for the applications you are distributing. For information, see **Setting Construct Spectrum Security Options**, page 95.

Note: To make a mirror image copy of the Spectrum system files, use ADAULD to unload and ADALOD to reload.

Data Transfer Utilities

The data transfer utilities allow you to copy domains and groups between one Spectrum system file and another. These utilities copy data either to a workfile or PC file, depending on the definition of the `WORK` parameter in your `NATPARM`. They also load data either from a workfile or PC file, depending on your `NATPARM`.

The following sections describe how to use the data transfer utilities in the Administration subsystem. For information about using the utilities in batch, see **Using the Data Transfer Utilities in Batch Mode**, page 141.

Note: To transfer data between version 4.1.1 files and higher files, invoke the utilities directly rather than use the Administration subsystem menus.

To invoke a utility, enter one of the following commands at the Next prompt:

Utility	Command
Export Domains utility	BS_EXDOM
Import Domains utility	BS_IMDOM
Export Groups utility	BS_EXGRP
Import Groups utility	BS_IMGRP

Transferring Domains

To transfer domains from one Administration subsystem file to another, use the Export Domains and Import Domains utilities. These utilities copy the following:

- All domain definitions
- All application service definitions to which the domain is attached, which includes the object descriptions, methods, and subprogram proxy names attached to the application service definition
- All steplib chains attached to the domain

Accessing the Domain Transfer Utilities

- To access the Domain transfer utilities:
 - 1 Enter “AA” in the Function field on the Construct Spectrum Administration Subsystem main menu.
The Application Administration main menu is displayed.
 - 2 Enter “MM” in the Function field on the Application Administration main menu.
The Application Administration Maintenance menu is displayed.
 - 3 Enter “DT” in the Function field on the Application Administration Maintenance menu.
The Transfer Domains menu is displayed:

```

BS_XFER2   ***** Construct Spectrum Administration Subsystem *****   CDLAYMNI
Apr 02                - Transfer Domains Menu -                          10:51 AM

                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

```

Transfer Domains Menu

- 4 Enter the code for the utility you want to access in the Function field.
The window for that utility is displayed.

Exporting Domains

➤ To export domains:

- 1 Enter “XD” in the Function field on the Transfer Domains menu. The Export Domains window is displayed:

```

Export Domains
BS_EXDOM                                BSEXDOM0

Domains
From..... _____
To..... _____
Enter-PF1---PF2---PF3---PF4---PF5-
confm help  retrn quit  deflt flip

```

Export Domains Window

- 2 Specify the domain or range of domains as follows:
 - To export one domain, type the domain name in the From field.
 - To export a range of domains, type the beginning and ending values in the From and To fields, respectively.
 - To export all domains with the same initial characters, type the characters followed by an asterisk (*) in the From field.
 - To export all domains up to and including a specific domain, type the domain in the To field.
- 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 in which to load the data.

When the data is exported, a confirmation message is displayed. If the export operation was not successful, a message describing the problem is displayed.

Importing Domains

➤ To import domains:

- 1 Log onto the library to which you want to import the domains.
- 2 Invoke the Administration Subsystem.
- 3 Access the Transfer Domains menu.

- 4 Enter “ID” in the Function field on the Transfer Domains menu.
The Import Domains window is displayed:

```

BS_IMDOM                                Import Domains                                BSIMDOM0

      Import Domains.....:  X
      <OR>
      Only scan input file and generate report.:  _

      Replace steplib chains.....:  X
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9-
      help  retrn quit          flip

```

Import Domains Window

- 5 Specify the options to use:
- To import data to the current library, mark Import Domains.
 - To only create a report of the data to be imported, mark the Only scan input file and generate report field. The report lists the steplib chains, domains, and interfaces (application service definitions). The report is written to the default device for Report 1 as set up by your system administrator.
 - The Replace steplib chains field is marked by default, indicating that steplib chains in the target environment with the same names as those being imported are replaced. If you do not want this option, remove the X from this field.
- 6 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 workfile, the import utility uses that workfile.

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.

Transferring Groups

To copy groups and their users from one Administration subsystem file to another, use the Export Groups and Import Groups utilities.

Accessing the Group Transfer Utilities

- To access the group transfer utilities:
 - 1 Enter “SA” in the Function field on the Construct Spectrum Administration Subsystem main menu.
The System Administration main menu is displayed.
 - 2 Enter “MM” in the Function field on the System Administration main menu.
The System Administration Maintenance menu is displayed.
 - 3 Enter “DT” in the Function field on the System Administration Maintenance menu.
The Transfer Groups menu is displayed:

```

BS_XFER      ***** Construct Spectrum Administration Subsystem *****      CDLAYMN1
Apr 02              - Transfer Groups Menu -                                10:30 AM

                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

```

Transfer Groups Menu

- 4 Enter the code for the utility you want to access in the Function field.
The window for that utility is displayed.

Exporting Groups

➤ To export groups and users:

- 1 Enter “XG” in the Function field on the Transfer Groups menu. The Export Groups window is displayed:

```

Export Groups
BS_EXGRP                                BSEXGRP0

Groups
From..... _____
To..... _____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---
help retn quit flip

```

Export Groups Window

- 2 Specify the group or range of groups to be exported as follows:
 - To export one group, type the group name in the From field.
 - To export a range of groups, type the beginning and ending values in the From and To fields, respectively.
 - To export all groups with the same initial characters, type the characters followed by an asterisk (*) in the From field.
 - To export all groups up to and including a specific group, type the group name in the To field.

- 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 transferred.

After the data is exported, a confirmation message is displayed. If the operation was not successful, a message describing the problem is displayed.

Importing Groups

➤ To import groups:

- 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 the Function field on the Transfer Groups menu.
The Import Groups window is displayed:

```

BS_IMGRP                                Import Groups                                BSIMGRP0

      Import Groups.....: X
      <OR>
      Only scan input file and generate report.: _
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---
      help  retrn quit          flip

```

Import Groups Window

- 5 Specify the options you want to use as follows:
- To import data to the current library, mark Import Groups.
 - To only create a report of the data to be imported, mark the Only scan input file and generate report field. The report lists the groups and users in the data file. The report 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 workfile, the import utility uses the data from that workfile.

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.

Using the Data Transfer Utilities in Batch Mode

You can use the following JCL examples to call the data transfer utilities in batch.

Exporting Domains

```

. . .
//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 SYSSPEC
BS_EXDOM <from domain> <to domain>

```

Importing Domains

```

. . .
//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 SYSSPEC
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.

Exporting Groups

```

. . .
//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 SYSSPEC
BS_EXGRP <from group> <to group>

```

Importing Groups

```

. . .
//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 SYSSPEC
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 Workfile

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 Construct Spectrum Import Groups utility. To help create this module, Spectrum supplies the BSSI_EX1 module in the SYSSPEC library as a sample template for a program that uses the workfile definitions.

The workfile 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 table with regard to data values and relative positioning within the record.

Header Record

This record must be the first record in the workfile. It identifies the version of the utility that created the file and the version of the Spectrum 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	4.5.1	10	3	12	Version of export utility
Data Version	0451	4	13	16	Version of Spectrum 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 Spectrum 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 Spectrum Users table. When creating an export file for both groups and users, ensure that Group records are copied to the workfile before User records.

Note: The Groups Export utility copies User records to the workfile in an encrypted format. If you write your own export utility to create an export workfile, 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

Field	Value	Length	Start Position	End Position	Description (continued)
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 workfile 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

APPENDIX A — PARSE USER PARAMETERS

Construct Spectrum supplies the BSSPARMN routine, which parses the keywords specified in the Service Start parameters fields on the Maintain Services panel (for information, see **Defining Service Start Parameters and Routines**, page 48).

BSSPARMN uses the BSSPARMA and CDPDA-M parameter data areas, which define the interface to the routine. These data areas are described in the following sections.

BSSPARMA Parameter 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 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 screen, 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 screen after parsing them. This allows you to supply the name of a test or trace facility.

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 <i>n</i> , if the <i>n</i> th 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 SYSSPEC application. 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=SYSSPEC"'
*
* 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

```



APPENDIX B — ADDITIONAL ERROR-HANDLING

To enhance the core error-processing, Construct Spectrum supplies the SPSERRN user exit. This user exit allows you to add special error-processing, such as sending notification of errors to the operator.

SPSERRN provides the following information:

- Error number
- Line number
- Status
- Program name
- Program level

SPSERRN uses the CDERRLDA parameter data area.

Note: Do not STOP or TERMINATE processing in this subprogram or any associated modules. To ensure complete tracking of internal Spectrum information, control should return to the Spectrum error handler.

Note: If you use SPSERRN to code additional error processing, be sure to back up the program before upgrading Construct Spectrum in the future. You can move the program back after installation is complete.



Construct Spectrum Administration _____

INDEX

A

- ACF2 security
 - using EntireX, 37
- ActiveX business object
 - role in Construct Spectrum architecture, 27
- Administration subsystem
 - See Construct Spectrum Administration subsystem
- Application administration, 28
- Application service definition
 - and Spectrum security, 96
 - role in application administration, 29
- Architecture
 - Construct Spectrum applications, 21
- ASCII character set
 - translation, 23–24
- Attach server
 - activation policy, 46
 - and SUBTASKB service start routine, 51
 - launching services in batch or as subtasks, 46
 - linking services to, 46
 - runtime error-handling, 46
 - Spectrum Service Manager, 116
- Attach service
 - and the START command, 47
 - defining, 42
 - description, 40
 - name supplied with Construct Spectrum, 45
 - specifying name, 45
 - starting as a started task, 131
- Attribute file
 - Entire Broker, 57

B

- Batch
 - initiating services using, 47
- BATCHJOB service start routine, 49
 - and JCL keywords, 49
 - example, 51
 - NATRJE, 48
- Broker Driver program
 - Automatic logon, 124
 - Conversation ID, 124
 - functions, 113
 - Security token, 124
 - Service, 124
 - using, 122
- Broker ID, 44
- BS_SYNCN exit module, 106
- BS_USEC* security exits
 - description, 104
- BS_USEC1 security exit, 105
- BS_USEC2 security exit, 105
- BSSI_EX1 module
 - Export Users and Groups workfile, 143
- BSSPARMA parameter data area, 147
- BSSPARMN routine, 147

C

- Callnat service, 41
- CALLNAT simulation, 25
- CA-TOP SECRET security
 - using EntireX, 37
- CDERRLDA parameter data area, 151
- CDPDA-M parameter data area, 149
- Change Password program
 - functions, 113
 - using, 118

- Character UI
 - role in Construct Spectrum architecture, 22
- Cleanup queue, 111
- Client service
 - defining, 42
 - description, 40
 - displaying in Manage Services panel, 58
- Client tools
 - description, 113
- Command service, 41
 - CALLNAT, 41
 - ENVIRONMENT, 41
 - ERROR, 41
 - PING, 41
 - SHUTDOWN, 41
 - STEPLIBS, 41
 - syntax, 41
 - SYSPROF, 41
 - TRACE, 41
- Common server
 - startup interface, 47
- Common start facility
 - SPSSTART common start facility, 40
- Communication logs
 - viewing from Manage Services panel, 60
- Compression/decompression of data, 23–24
- Construct Spectrum
 - Administration subsystem
 - uses, 28
 - using menus with Natural Security, 36
 - documentation, 17
 - runtime environment, 126
 - security cache, 30
- Construct Spectrum Administration documentation
 - how to use, 13
 - layout, 11
 - purpose and structure, 11
- Construct Spectrum Administration subsystem
 - application administration, 28
 - deploying, 126
 - invoking, 31
 - using a hybrid security service, 38
 - using EntireX security, 37
 - using Natural Security, 34
- Construct Spectrum SDK
 - documentation, 16
- Construct Spectrum security
 - accessing log query panels, 90
 - accessing queries and logs, 89
- Construct Spectrum service
 - attach service, 40, 42
 - See also Attach service
 - client service, 40, 42
 - See also Client service
 - command service, 41
 - See also Command service
 - debugging, 61
 - definition, 39
 - initiating, 60, 63
 - main service, 41
 - See also Main service
 - names supplied with Construct Spectrum, 45
 - pinging, 61, 64
 - refreshing, 61, 64
 - server service, 40, 42
 - See also Server service
 - shutting down, 61, 65
- Control record
 - accessing log, 74
 - enabling Natural Security, 73
 - Entire Net-Work timeout, 74
 - modifying control data, 73
 - security exit, 73
 - uses, 71
- Conventions
 - typographical
 - used in this documentation, 15
- Conversation factory
 - Spectrum Service Manager, 116
- Courses
 - related Natural Construct, 18

D

Data compression/decompression, 23–24

Data encryption/decryption, 23

Data transfer utilities, 135

- batch mode, 141
- export
 - creating, 143
- export domains in batch, 141
- Export Domains utility, 135, 137
- export groups in batch, 142
- Export Groups utility, 140
- Export Users and Groups workfile, 143
- import domains in batch, 142
- Import Domains utility, 135, 137
- import groups in batch, 142
- Import Groups utility, 140

Data translation, 23–24

DEBUG command, 60–61

Debug library and file, 84

Debugging information

- system functions, 24

Decryption/encryption of data, 24

Deploying

- Construct Spectrum Administration subsystem
 - attach service, 131
 - copying system files, 130
 - data transfer utilities, 135
 - dispatch service, 133
 - example, 127
 - Natural Security, 133
 - prerequisite software, 129
 - security service, 133
 - Spectrum system file database, 131
 - transferring domains, 134
 - transferring group and user tables, 134
- Construct Spectrum application, 126

Description text box

- Spectrum Service Manager, 115

Disabled security status, 103

Dispatch service

- encapsulation of Entire Broker calls, 25
- in program parameters, 56
- name supplied with Construct Spectrum, 45
- role in Construct Spectrum architecture, 23
- security mode, 56

- Spectrum Service Manager, 116
- system functions, 24

Documentation

- related Construct Spectrum, 17
- Software Development Kit (SDK), 16
- related Natural Construct, 16–17

Domain

- exporting, 137
- exporting in batch, 141
- importing, 137
- importing in batch, 142
- role in application administration, 29
- role in Construct Spectrum security, 76
- role in deploying, 134
- transfer utilities
 - accessing, 136
 - transferring, 135

E

EBCDIC character set

- translation, 23–24

Enabled security status, 103

Encryption/decryption of data, 23–24

ENDJOB command, 60

- description, 62

ENDSERVICE command, 60

- description, 62

Entire Broker

- attribute file
 - changing, 57
- encapsulation of calls by SDC, 25
- parameters
 - Spectrum Service Manager, 116
 - role in Construct Spectrum architecture, 23, 25, 27

Entire Broker service settings, 43–44

- attach service name, 45
- broker ID, 44
- server class, 44
- server name, 44
- service, 45
- user ID and password, 45

Entire Net-Work timeout, 74

EntireX security

- using with Construct Spectrum, 37

ENVIRONMENT command, 60

- description, 62

Environment service, 41
Error service, 41
Error-handling
 and attach server, 46
 CDERRLDA PDA, 151
 SPSERRN user exit, 151
 system functions, 24
ETB02150148 error
 description, 114
Export Domains utility, 135
Export Users and Groups workfile, 143
 Group record, 144
 Header record, 143
 User record, 144

G

Granted security status, 102
Group
 associating with user
 in Natural Security, 86
 in Spectrum security, 84
 defining
 using Natural Security, 80
 using Spectrum security, 79
 exporting, 140
 exporting in batch, 142
 importing, 140
 importing in batch, 142
 role in Construct Spectrum security, 30, 76
 transferring, 134, 139
Group transfer utilities
 accessing, 139
GUI dialog
 role in Construct Spectrum architecture, 26

H

Hybrid security service
 using with Construct Spectrum, 38

I

IIS
 role in Construct Spectrum architecture, 27
Import Domains utility, 135
Information logs, 28
INITIATE command, 60, 63
Internet/intranet
 role in Construct Spectrum architecture, 27

J

JCL keyword
 and BATCHJOB service start routine, 49
JOB-NAME JCL keyword
 BATCHJOB service start routine, 49
JOB-PREFIX keyword
 BATCHJOB service start routine, 49
JOB-TRANSACTION JCL keyword
 BATCHJOB service start routine, 50

L

Library image file
 simulated PDAs in, 25
LIBRARY JCL keyword
 BATCHJOB service start routine, 50
Library Maintenance/Link Users window
 accessing, 88
LIF, see Library image file
Log Query panels
 accessing, 69
Logical File 135
 Construct Spectrum system file, 130
Logical File 136
 Construct Spectrum system file, 130
Logs
 information, 28

M

- Main service
 - description, 41
- Maintain Control Record window
 - accessing, 72
 - enabling Natural Security, 73
 - Entire Net-Work timeout, 74
 - modifying control data, 73
 - security exit, 73
- Maintain Group Profile window
 - accessing, 80
 - using, 80
- Maintain Group Table panel
 - accessing, 79
- Maintain Object Security by Group panel, 99
 - Clear action, 99
- Maintain Object Security panels
 - accessing, 97
- Maintain Services panel 1
 - attach, 46
 - Description field, 44
 - Entire Broker service settings, 43–44
 - attach service name, 45
 - broker ID, 44
 - server class, 44
 - server name, 44
 - service, 45
 - user ID and password, 45
 - Name field, 44
 - service start parameters, 43, 47–48
 - service start routine, 48
- Maintain Services panel 2
 - default language, 54
 - description, 54
 - program parameters, 56
 - attach client, 56
 - dispatch service, 56
 - Natural program name, 56
 - service timeout, 55
 - service type, 54
 - unique ID, 55
- Maintain Services panels
 - accessing, 42
- Maintain User Profile window
 - accessing, 85
 - using, 85
- Maintain User Table panel
 - accessing, 83
 - debug file name, 84
 - debug library, 84
 - groups, 84
- Maintain User's Groups window
 - accessing, 87
 - using, 87
- Manage Services panel
 - accessing, 58
 - DEBUG command, 60
 - displaying client services, 58
 - ENDJOB command, 60
 - ENDSERVICE command, 60
 - ENVIRONMENT command, 60
 - INITIATE command, 60
 - Last command field, 59
 - PING command, 61
 - purpose, 58
 - REFRESH command, 61
 - Repeat command field, 59
 - Replica ID field, 59
 - Respondent field, 60
 - Service field, 59
 - Service type field, 59
 - SHUTDOWN command, 61
 - STEPLIBS command, 61
 - SYSPROF command, 61
 - viewing Communication logs, 60
- Message-handling
 - system functions, 24
- Method
 - role in application administration, 29
 - role in Construct Spectrum security, 76
- Microsoft Internet Information Server
 - role in Construct Spectrum architecture, 27

N

- Name text box
 - Spectrum Service Manager, 115
- NATPARM JCL keyword, 52
 - BATCHJOB service start routine, 50
- Natural Construct
 - documentation and course information, 16
- Natural profile
 - adding, 57

Natural Security
 accessing query panels, 91
 and SUBTASKB service start routines, 52
 configuring for Construct Spectrum, 34
 enabling on the control record, 73
 interface to restricted libraries, 35
 Library Maintenance/Link Users window, 88
 List Groups for a User window, 93
 List Groups window, 92
 List Users window, 93
 Maintain Group Profile window, 80
 Maintain User Profile window, 85
 Maintain User's Groups window, 87
 role in deploying Construct Spectrum Administration subsystem, 133
 using with Construct Spectrum, 34

Natural subprogram
 role in Construct Spectrum architecture, 22

No Privileges security status
 description, 102

NSCUSEX1 module
 configuring Natural Security environment, 34

O

Object
 role in Construct Spectrum security, 76

Optimistic security checking, 105

P

Parameter data areas
 simulation by Spectrum Dispatch Client, 25

PASSWORD JCL keyword
 BATCHJOB service start routine, 50

Pessimistic security checking
 description, 105

PING command, 61, 64

Ping service, 41

PROGPARM JCL keyword
 BATCHJOB service start routine, 50

Program parameters for services, 56
 attach client, 56
 dispatch service, 56

Q

Query Communications Logs panel
 accessing, 67

Query panels
 accessing, 68

R

RACF security
 using EntireX, 37

REFRESH command, 61, 64

Remote Clock program
 command line parameters, 120
 functions, 113
 Spectrum Dispatch Client, 120
 using, 119

Reset
 Spectrum security cache, 37

Response time-out
 Spectrum Service Manager, 117

Revoked security status, 102

Runtime environment, 126

S

Security administration
 activities, 30

Security cache, 30
 synchronizing using BS_SYNCN, 106
 synchronizing with cleanup queue, 111

Security exit, 73
 BS_USEC1, 105
 BS_USEC2, 105
 BS_USECO, 104
 Construct Spectrum, 96
 using security exit modules, 104

Security for Group-Domain window
 description, 100

Security for Object window
 Clear action, 101
 description, 101

- Security mode
 - for dispatch service, 56
 - Spectrum Service Manager, 117
- Security service
 - name supplied with Construct Spectrum, 45
 - role in Construct Spectrum architecture, 23
 - using a hybrid, 38
- Security settings
 - adding group-domain combination, 98
 - defining, 97
 - options for group-domain combination, 99
 - options for methods, 101
 - options for objects, 100
 - purging group-domain combination, 98
 - statuses
 - Disabled, 103
 - Enabled, 103
 - Granted, 102
 - No Privileges, 102
 - Revoked, 102
- Server class, 44
- Server name, 44
- Server service
 - defining, 42
 - description, 40
- Server startup process
 - attach server, 46
 - automating with main and command services, 41
 - initiating services as subtasks, 47
 - initiating services in batch, 47
 - initiating services online, 47
 - START command, 47
- Service
 - in Entire Broker service settings, 45
- Service ID
 - Spectrum Service Manager, 115
- Service Properties window, 115
- Service Start parameters, 43
 - BSSPARMA PDA, 147
 - CDPDA-M PDA, 149
 - parsing with BSSPARMN, 147
- Service start routine
 - BATCHJOB, 48–49
 - generating trace output, 53
 - SUBTASK, 48
 - SUBTASKB, 51
 - STACK parameter, 51
 - using, 48
 - writing to print and work file, 52
- Service startup process
 - service start parameters, 48
 - service start routine, 48
 - under Natural Security, 52
- Service timeout, 55
- Service type
 - Spectrum Service Manager, 116
- Services
 - attach
 - starting as a started task, 131
- SHUTDOWN command, 61, 65
- Shutdown service, 41
- Shutting down services
 - all replicas, 65
 - all servers, 65
 - single replica, 65
- Spectrum administration
 - role in Construct Spectrum architecture, 23
- Spectrum Control Record Log
 - accessing, 74
- Spectrum Dispatch Client
 - CALLNAT simulation, 25
 - database transaction control, 26
 - encapsulation of Entire Broker calls, 25
 - role in Construct Spectrum architecture, 25, 27
- Spectrum dispatch service
 - in program parameters, 56
 - role in Construct Spectrum architecture, 23
 - role in deploying Construct Spectrum Administration subsystem, 133
 - Spectrum Service Manager, 116
 - system functions, 24
- Spectrum security
 - accessing log query panels, 90
 - accessing queries and logs, 89, 107
 - and version of application service definition, 96
 - disabled status, 96
 - enabled status, 96
 - for applications, 95
 - levels of security, 96

Spectrum security service
 role in Construct Spectrum architecture, 23
 role in deploying Construct Spectrum Administration subsystem, 127, 133

Spectrum service
 attach service, 40, 42
 See also Attach service
 client service, 40, 42
 See also Client service
 command service, 41
 See also Command service
 copying definitions, 117
 debugging, 61
 definition, 28
 initiating, 60, 63
 main service, 41
 See also Main service
 pinging, 61, 64
 pinging using the Spectrum Service Manager, 113
 refreshing, 61, 64
 server service, 40
 See also Server service
 shutting down, 61, 65
 starting in batch mode
 under Natural Security, 35

Spectrum Service Manager
 functions, 113
 Service Properties window, 115
 Description, 115
 Entire Broker Parameters, 116
 Name, 115
 Response time-out, 117
 Security mode, 117
 Service ID, 115
 Service type, 116

SPSEERN user exit, 151

SPSSTART common start facility, 47

START command, 47
 specifying Natural program name, 56
 to initiate services in batch, 47

Starting services, 43

Steplib
 role in application administration, 29

STEPLIBS command, 61
 description, 66

Steplibs service, 41

Subprogram proxy
 role in Construct Spectrum architecture, 22
 Spectrum dispatch service, 23

Subtask
 initiating services as, 47
 of attach server, 46

SUBTASK service start routine
 CMTASK, 48

SUBTASKB service start routine, 51
 example, 52
 NATPARM keyword, 52
 STACK parameter, 51
 TRANSACTION keyword, 52

Synchronizing
 security cache, 106

Synchronizing the security cache with the cleanup queue, 111

SYSPROF command, 61
 description, 67

Sysprof service, 41

System administration tools
 accessing, 110

T

Timeout
 for service, 55

Timestamp service
 name supplied with Construct Spectrum, 45
 Remote Clock program, 119

Tools
 client, 113
 system administration
 accessing, 110

TRACE command, 53
 example, 53

Trace output
 generating, 53

Trace service, 41

Translation of character sets, 23–24

Typographical conventions
 used in this documentation, 15

U

User

- associating with groups
 - in Natural Security, 86
 - in Spectrum security, 84
- defining, 82
 - using Construct Spectrum security, 83
 - using Natural Security, 85
- defining debug library and file, 84
- role in Construct Spectrum security, 76
- transferring, 134

User ID

- role in security administration, 30

User ID and password

- in Entire Broker service settings, 45

USER-ID JCL keyword

- BATCHJOB service start routine, 50

V

Visual Basic business object

- role in Construct Spectrum architecture, 26

W

Web application

- role in Construct Spectrum architecture, 27

