



**Sybase Control Center for Sybase Unwired
Platform**

Sybase Unwired Platform 2.1

ESD #1

DOCUMENT ID: DC01092-01-0211-01

LAST REVISED: December 2011

Copyright © 2011 by Sybase, Inc. All rights reserved.

This publication pertains to Sybase software and to any subsequent release until otherwise indicated in new editions or technical notes. Information in this document is subject to change without notice. The software described herein is furnished under a license agreement, and it may be used or copied only in accordance with the terms of that agreement.

To order additional documents, U.S. and Canadian customers should call Customer Fulfillment at (800) 685-8225, fax (617) 229-9845.

Customers in other countries with a U.S. license agreement may contact Customer Fulfillment via the above fax number. All other international customers should contact their Sybase subsidiary or local distributor. Upgrades are provided only at regularly scheduled software release dates. No part of this publication may be reproduced, transmitted, or translated in any form or by any means, electronic, mechanical, manual, optical, or otherwise, without the prior written permission of Sybase, Inc.

Sybase trademarks can be viewed at the Sybase trademarks page at <http://www.sybase.com/detail?id=1011207>. Sybase and the marks listed are trademarks of Sybase, Inc. ® indicates registration in the United States of America.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world.

Java and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

Unicode and the Unicode Logo are registered trademarks of Unicode, Inc.

All other company and product names mentioned may be trademarks of the respective companies with which they are associated.

Use, duplication, or disclosure by the government is subject to the restrictions set forth in subparagraph (c)(1)(ii) of DFARS 52.227-7013 for the DOD and as set forth in FAR 52.227-19(a)-(d) for civilian agencies.

Sybase, Inc., One Sybase Drive, Dublin, CA 94568.

Contents

- Get Started1**
 - About Sybase Control Center for Unwired Platform1
 - Documentation Roadmap for Unwired Platform1
 - Unwired Platform Administration by Node2
 - Cluster Administration Overview3
 - Server Administration Overview4
 - Application and User Management Overview5
 - MBO Package Management Overview6
 - Security Administration Overview8
 - System Monitoring Overview8
 - Mobile Workflow Package Management Overview10
 - Starting and Stopping Sybase Control Center in Unwired Platform10
 - Cleaning Up the Flash Player Cache11
 - Getting Started with Production Clusters11
 - Getting Started After Installing12
 - Configuring Memory Usage14
 - Configuring the Automatic Logout Timer15
 - Manually Opening the Unwired Platform Console16
 - Adding or Updating Unwired Server Registration Properties17
 - Understanding the Sybase Control Center Interface18
 - User Interface Overview18
 - Perspectives20
 - Views21
 - Repository22
 - Sybase Control Center Console27
 - Sybase Control Center Security29

Administer	31
Clusters	31
Cluster-Affecting Configuration Changes	31
Copying and Pasting Properties	31
Configuring General Cluster Properties	32
Viewing Cluster Information	32
Checking System Licensing Information	33
Checking Cluster Status	34
Relay Server	34
Configuring Unwired Server to use Relay Server	35
Managing Configured Relay Servers	39
Relay Server Tab Reference	42
Unwired Server	42
Server List	43
Server Properties	45
Relay Server Outbound Enabler	59
Server Log	67
Domains	75
Creating and Enabling a New Domain	76
Deleting a Domain	77
Registering a Domain Administrator User	77
Assigning Domain Administrators to a Domain ..	78
Viewing Applications for a Domain	78
Viewing Application Connections for a Domain ..	79
Scheduling Domain-Level Cleanup	79
Domain Logs	83
Checking Client Application Logs	108
Connections	108
Configuring Domain Security	132
Security Configurations	136
Creating a Security Configuration	137
Assigning a Security Configuration to a Domain	166
Applications	166

Setting Up Application and User Connections ..	167
Application Creation	167
Application ID Overview	168
Modifying Application Properties	169
Deleting Applications	169
Application Connection Activation Options	169
Managing and Searching for Applications	170
Searching for Applications	171
Application Users	172
Application Connections	173
Application Connection Templates	177
Application Connection Properties	179
Deploy	185
Packages	185
Deploying Packages	185
Package Management	190
Subscription Management	203
Reviewing MBO History	208
Reviewing Operation History	209
DOE-C Packages	209
Setting the Bulk Load Timeout Property	209
Checking and Resolving DOE-C User Failures .	210
Package Subscription Properties	211
Mobile Workflows	211
Enabling and Configuring the Notification	
Mailbox	211
Configuring a Mobile Workflow Package	212
Deploying a Mobile Workflow Package	218
Monitor	221
Monitoring Usage	221
System Monitoring Overview	222
Monitoring Configuration	223
Configuring Monitoring Performance Properties	
.....	224
Monitoring Profiles	225

Creating and Enabling a Monitoring Profile	225
Monitoring Data	227
Reviewing System Monitoring Data	227
Purging Monitoring Data	228
Exporting Monitoring Data	228
Searching Monitoring Data	229
Monitoring Data Categories	229
Troubleshoot the Sybase Control Center	251
Using Sybase Control Center to Troubleshoot Unwired Platform	251
Collecting Administration Performance Data for Troubleshooting	252
Sybase Control Center Management Tier Issues	253
Launching SCC Results in Rounded Rectangle Box or Empty Console Screen	253
Sybase Control Center Windows Service Fails to Start	254
Sybase Control Center Windows Service Deleted	255
Sybase Control Center Fails to Start	255
Second Sybase Control Center Fails to Start ...	255
Login Invalid in Sybase Control Center	256
Login Fails in Sybase Control Center	256
Administrator Account is Locked	256
Browser Refresh (F5) Causes Logout	257
Stale Version of Sybase Control Center After Upgrade	257
Sybase Control Center Reports Certificate Problem	257
Previous Administrator Credentials Used	258
Security Error Triggered When Connecting to SCC from Remote Browser	258
Administrator Login Passes When Provider Is Not Available	258

Host Name of Registered Resource Changed But Is Not Updated	259
Management Issues with Clustered Data Tiers	259
Poor Sybase Control Center Performance after Upgrade	259
Sybase Control Center Communication with Unwired Server Fails	259
Platform Component Monitoring Issues	260
Monitoring Data Does Not Appear in History Tab	261
Domain Log Data Does Not Appear in History Tab	261
Previously Existing Monitoring Data No Longer Appears	262
Previously Existing Domain Log Data No Longer Appears	262
Server Tier Administration Issues	263
Server List Not Retrieved	263
Unwired Server Fails to Start	265
Error in Listing Application Connections and ADMIN_WEBSERVICE_INVOCATION_ERR OR in gateway.log	265
Starting or Restarting a Remote Server from Sybase Control Center Fails	266
Port Conflict Issues	267
Unexpected Listener Startup or Connection Errors	267
Refreshing Server Configuration Displays Only Partial Updates	268
Users Connect with Old Credentials	269
AuthorizationException Displays Instead of Status	269
Increasing Messaging Queue Counts Degrades Performance	270

Saving Server Configuration Fails Due to Certificate Validation Error	270
Unknown Server Error Message	270
Package Deployment and Management Issues	270
Exporting or Deploying Large Packages Fails ...	270
Invalid DOE-C User Error for an SAP Server Connection	271
Application and Application User Management Issues	272
Wrong Application for Code Error	272
User Name of Registered Application Connection Not Displayed	273
Internal Server Error When Clicking Applications	273
Glossary	275
Glossary: Sybase Unwired Platform	275
Index	287

Get Started

Set up Sybase® Control Center.

About Sybase Control Center for Unwired Platform

Sybase Control Center provides a single comprehensive Web administration console to configure and manage Sybase products and their components.

Sybase Control Center combines a modular architecture, a rich administrative console, agents, common services, and tools for managing and controlling Sybase products. Unwired Platform is one of many Sybase products that use Sybase Control Center as its management and administrative tool.

As part of an Unwired Platform installation, Sybase Control Center can be used in three ways:

- In a personal development environment, developers may act as administrators to set up a personal testing environment. Development administrators use Sybase Control Center to deploy and configure packages, register messaging devices, and so on. No other additional configuration or administration may be required.
- In a distributed or shared development environment, administrators use Sybase Control Center to set up an Unwired Server, manage packages, manage devices, configure mobile workflow packages, as well as review server and domain logs, and monitoring-related data.
- In a production environment, administrators use Sybase Control Center on a regular basis to perform the same tasks described for a shared development environment. They also configure the operation of Unwired Servers, and administer day-to-day activities of the production environment. Administrators must also routinely monitor the overall health and performance of the system, which may include clusters and domains.

Documentation Roadmap for Unwired Platform

Sybase® Unwired Platform documents are available for administrative and mobile development user roles. Some administrative documents are also used in the development and test environment; some documents are used by all users.

See *Documentation Roadmap* in *Fundamentals* for document descriptions by user role. *Fundamentals* is available on the Sybase Product Documentation Web site.

Check the Sybase Product Documentation Web site regularly for updates: access <http://sybooks.sybase.com/nav/summary.do?prod=1289>, then navigate to the most current version.

Unwired Platform Administration by Node

The left navigation pane in the Sybase Control Center for Unwired Platform console displays a tree of administrable features in the form of nodes, some of which can be expanded to reveal a more granular view of the cluster environment. These nodes let you manage and configure the main components of Unwired Platform.

Clicking nodes allows you to administer the following features through Sybase Control Center. However, be aware of the following dependencies:

- There are two administration roles. Users with the platform administration role have access to all nodes. Users with the domain administrator role see only the "Domains" nodes for their assigned domains.
- You must have the correct Unwired Platform version and license for these nodes to be functional when they are visible. For example, Sybase Mobile Sales and Sybase Mobile Workflow products may not have all the same functionality as Sybase Unwired Platform.

Node	Purpose
Cluster	View general cluster properties and access the server list for the cluster, and Relay Server configurations on the cluster level.
Domains	Add, delete, enable, and disable domains. Expand this node to manage the security, package, role mappings, cache group, synchronization group, subscription, and connection configurations for each domain. You can also expand the Applications subnode to see the applications and application connections managed from the domain.
Servers	View the list of servers, their properties, and their statuses. Expand this node to manage individual Unwired Servers, to configure properties and logs, and to apply pending changes.
Applications	Add, view, delete, and edit applications, application users, application connections, and application connection template operations as part of application activation.
Security	Add, view, edit, and delete domain administrators. Add or delete a security configuration. Each security configuration contains one or more security providers for authentication, authorization, and auditing. Once configured, security configurations can be assigned to domains and then mapped to one or more packages, depending on the requirements for each.
Workflows	Deploy and manage mobile workflow packages and configure the notification mailbox. Deployed mobile workflow packages are listed below this node. Use the individual mobile workflow nodes to manage mobile workflow package properties, matching rules, context variables, error logs, application connections, and, optionally, queue items.

Node	Purpose
Monitoring	Create and manage settings for monitoring security, replication synchronization, messaging synchronization, device notification, data change notification, queue, package, user, and cache activities.

Cluster Administration Overview

The goal of cluster administration is to ensure that clusters and servers work smoothly, and scale over time. By default, the Unwired Platform is installed as a one-node cluster. The one-node cluster is supported in development or test environments. Production deployments of Unwired Platform are likely to require multiple nodes. Cluster administration is mostly a nonroutine administration task.

See *Installation Guide for Runtime > System Deployment Overview*.

Table 1. Cluster administration tasks

Task	Frequency	Accomplished by
Installing the cluster	One-time installation per cluster	Unwired Platform installer
Setting up relay servers	One-time initial installation and configuration; occasionally adding servers to the cluster	Manual installation; manual set-up using configuration files.
Suspending and resuming server nodes	On demand, as required	Sybase Control Center
Setting cluster properties, including cache database settings, monitoring database setup, and so on	Once, or as cluster changes require	Manual configuration using files and .BAT scripts.
Set up a shared data folder to share cluster synchronization content as an alternative method for keeping secondary nodes in sync.	Once, or as cluster changes require	Sybase Control Center
Administering the runtime databases	Routine to ensure that the database server is monitored and backed up, that there is sufficient space for Unwired Platform metadata and cached data tables, and that performance is within acceptable limits (performance tuning)	Established processes and command line utilities. Consult with your database administrator. See <i>System Administration > Operation Maintenance</i> .

Task	Frequency	Accomplished by
Reviewing licensing information, including total licensed devices and currently used licenses count	Occasional, or as device user registration and deregistration occurs	Sybase Control Center. See <i>System Administration > Operation Maintenance</i> .

Server Administration Overview

The goal of server administration is to ensure that Unwired Server is running correctly and that it is configured correctly for the environment in which it is installed (development or production). Server administration is mostly a one-time or infrequent administration task.

Table 2. Server administration tasks

Task	Frequency	Accomplished by
Installing the server	One-time installation per server	Unwired Platform installer.
Configuring the server to: <ul style="list-style-type: none"> • Set the replication and messaging synchronization ports, as well as communication ports for administration and DCN • Create security profiles for secure communication • Set up secure synchronization • Configure replication and messaging push notifications • Tune server performance 	Postinstallation configuration with infrequent tuning as required	Sybase Control Center for Unwired Platform.
Manage the outbound enabler configuration for Relay Server. <ul style="list-style-type: none"> • Configure Relay Server properties • Manage certificates • View logs • Configure proxy servers for outbound enabler 	Postinstallation	Sybase Control Center for Unwired Platform.
Setting server log file settings and subsystem log levels	Once, unless log data requirements change	Sybase Control Center for Unwired Platform.

Application and User Management Overview

The goal of application management is to register an application to Unwired Server as an entity, create an application template that specifies application connection details for a user, and activate applications either manually or automatically.

Table 3. Application and user management tasks

Task	Frequency	Accomplish by using
Create new applications to register application entities with Unwired Server. A default application template is created automatically. Modify and delete applications as part of application life cycle.	As required	Sybase Control Center for Unwired Platform with Applications node, and Applications tab.
Create or modify application connection templates to specify details for native, workflow, and proxy application connections.	As required	Sybase Control Center for Unwired Platform, with Application node, and Application Connection Templates tab.
For applications that need to be registered manually, register an application connection to associate an application connection with a user. This is not necessary for applications that are registered automatically.	As required	Sybase Control Center for Unwired Platform, with Application node, and Application Connections tab.
View activated users, once they have logged in with the activation code. Users must either supply the activation code manually, or the device client supplies the activation code automatically as coded.	As required	Sybase Control Center for Unwired Platform with the Application node, and Application Users tab.
Create a new activation code for a user whose code has expired.	As required	Sybase Control Center for Unwired Platform, with Application node, and Application Connections tab.
Review registered application connections and users, delete application connections to free licenses, delete application connections to remove users from the system	As required	Sybase Control Center for Unwired Platform with the Applications node.
Manage subscriptions	As required	Sybase Control Center for Unwired Platform with the Packages node.

Information and guidelines:

Note: Application connection is not used for native replication applications at this time.

Get Started

- Two activation options are available for onboarding, which refers to the process of activating an authentic device client, user, and application entity, as a combination, in Unwired Server:
 - Automatic activation – requires a user to present credentials to use the application on a supported device client.
 - Manual activation – requires the user to present the activation code upon log on to a supported device client. The system administrator establishes an activation code when registering the application connection for the user.
- Native messaging applications can only be activated manually.
- Application template are used for automatic activation. Therefore, when setting up the application template for automatic registration, be sure to set up the security configuration, domain, the application ID, and automatic registration enabled properties in application settings. Those are used for automatic application registration.

When a client application connects to the server with its application ID and credentials, and requests automatic registration, the application ID is used to look up a matching template. If that template allows automatic registration (the Automatic Registration Enabled property is set to true), the security configuration in the template is used to validate the credentials. Upon successful validation of those credentials, the user identity is registered in the Unwired Server. The client application may also include the security configuration as part of the username (user@securityconfiguration) and in that case, the security configuration (in addition to application ID) is used to look up a matching template. If no or multiple templates are detected, the registration request fails.

- Supported device client activation options:

Device Client Type	Automatic Registration	Manual Registration
Workflow messaging	X	X
Native messaging		X
Online Data Proxy	X	X

MBO Package Management Overview

The goal of mobile business object (MBO) package management is to make MBOs available to device users. MBO package management typically requires a one-time deployment and configuration, except for ongoing subscription management for messaging and Data Orchestration Engine connector (DOE-C) packages.

Packages contain MBOs that are deployed to Unwired Server to facilitate access to back-end data and transactions from mobile devices. Package types include UNIFIED packages, and SAP® DOE-C packages.

A package, along with its current settings for cache groups, role mappings, synchronization groups, connections, and security configuration, can be exported to an archive and imported back into Sybase Control Center for backup or to facilitate a transition from a test environment to a production environment.

Table 4. MBO package management tasks

Task	Package type	Frequency	Accomplish by using
Deploy packages to a development or production Unwired Server	UNIFIED	Once, unless a new version becomes available	Sybase Control Center for Unwired Platform with the domain-level Packages node
Control user access by assigning security configurations for each package, and mapping roles if fine-grained authorization is enforced through logical roles	UNIFIED	Once, unless security requirements of the package change	Sybase Control Center for Unwired Platform with the domain-level Packages node
Set up the package cache interval and cache refresh schedule (for getting data updated on the Unwired Server from the data source)	UNIFIED	Once, unless data refreshes need to be tuned	Sybase Control Center for Unwired Platform with the domain-level Packages node
Manage subscriptions (UNIFIED, and DOE-C), synchronization groups (UNIFIED), and device notifications (UNIFIED) to customize how updated data in the cache is delivered to the device user	Varies	Periodic, as required	Sybase Control Center for Unwired Platform with the domain-level Packages node
Export or import an MBO package	UNIFIED	On-demand, as required	Sybase Control Center for Unwired Platform with the domain-level Packages node
Review current/historical/performance metrics	All	Routine	Sybase Control Center for Unwired Platform with the Monitor node (available only to administrators)
View asynchronous operation replays for the selected package	Replication, UNIFIED	Periodic, as required	Sybase Control Center for Unwired Platform with the domain-level Packages node. However, asynchronous operation replays must first be enabled at the cluster level. See <i>Viewing Asynchronous Operation Replays</i> in Sybase Control Center online help.

Security Administration Overview

Perform security administration tasks to establish rules for the protection of enterprise and administrative data and transactions.

Unwired Server coordinates data between enterprise information server (EIS) data sources and device clients, meaning that transferred information is often proprietary, confidential, or private. Therefore, the data and communication streams that carry information from Unwired Server to other components in the Unwired Platform must be protected.

Unwired Platform has several security layers that protect data and transactions. Administrators manage system and application authentication and authorization security configurations at the cluster level, and perform role mapping at the domain and package levels. By default, the 'admin' security configuration is used to authenticate and authorize all administrative users, including domain administrators. All domain administrator logins must be valid in the security repository configured for the 'admin' security configuration.

Platform administrators register domain administrators at the cluster level, and then assign them to a domain from the domain-level Security Configurations tab. Security configurations are assigned when domains are created, or subsequently, from the Domains node. Packages must also be mapped to a security configuration at deployment; role mapping can be configured at a later time.

Roles are used for MBOs and operations during development to indicate authorization requirements. These roles are enforced by Unwired Server. At deployment or after deployment, these logical roles can be mapped to physical roles to restrict which users have access to MBOs and operations. Roles assigned at the MBO level are separate from operation-level roles. However, package-level role mapping overrides domain-level role mapping. If the same package is deployed to multiple domains and associated with the same security configuration, then the domain-level role mapping is shared.

System Monitoring Overview

(Not applicable to Online Data Proxy) The goal of monitoring is to provide a record of activities and performance statistics for various elements of the application. Monitoring is an ongoing administration task.

Use monitoring information to identify errors in the system and resolve them appropriately. This data can also be shared by platform and domain administrators by exporting and saving the data to a .CSV or .XML file.

The platform administrator uses Sybase Control Center to monitor various aspects of Unwired Platform. Monitoring information includes current activity, historical activity, and general performance during a specified time period. You can monitor these components:

- Security log
- Replication synchronization

- Messaging synchronization
- System messaging queue status
- Data change notifications
- Device notifications (replication)
- Package statistics (replication and messaging)
- User-related activity
- Cache activity

To enable monitoring, platform administrators must set up a monitoring database, configure a monitoring data source or create a new one, and set up monitoring database flush and purge options. By default the installer created a monitoring database, however you can use another one if you choose.

To control monitoring, platform administrators create monitoring profiles and configurations, which define the targets (domains and packages) to monitor for a configured length of time. A default monitoring profile is created for you by the installer. Monitoring data can be deleted by the platform administrator as needed.

Table 5. System monitoring tasks

Task	Frequency	Accomplished by
Create and enable monitoring profiles	One-time initial configuration with infrequent tuning as required	Sybase Control Center for Unwired Platform with the Monitoring node
Enable domain logging	One-time setup with infrequent configuration changes, usually as issues arise	Sybase Control Center for Unwired Platform with the Domains > <DomainName> > Log node.
Review current/historical/performance metrics	Routine	Sybase Control Center for Unwired Platform with the Monitoring node
Identify performance issues	Active	Sybase Control Center for Unwired Platform with the Monitoring node
Monitor application and user activity to check for irregularities	Active	Sybase Control Center for Unwired Platform with the Monitoring node
Troubleshoot irregularities	Infrequent	Reviewing various platform logs
Purge or export data	On demand	Sybase Control Center for Unwired Platform with the Monitoring node

Mobile Workflow Package Management Overview

The goal of mobile workflow package management is to make mobile workflows available from the Unwired Server to device users. Mobile workflow package management typically requires a one-time deployment and configuration, except for ongoing package maintenance.

The mobile workflow application is a simple business process application that delivers functionality, such as sending requests and approvals through an e-mail application, to mobile device clients on supported device platforms, including Windows Mobile, iOS.

Table 6. Mobile workflow package management

Task	Frequency	Accomplish by using
Deploy mobile workflow packages	Once, unless a new version becomes available	Sybase Control Center for Unwired Platform with the Workflow node
Mobile workflow configuration that includes e-mail matching rules and context variables	Once	Sybase Control Center for Unwired Platform with the Workflow node
Device registration and user assignments to mobile workflow packages	Routine when new users or new devices are added	Sybase Control Center for Unwired Platform with the Workflow > <WorkflowName> node
Monitor users and errors	Routine	Sybase Control Center for Unwired Platform with the Monitor node

Starting and Stopping Sybase Control Center in Unwired Platform

Sybase Unified Agent is used to start and stop Sybase Control Center.

There are two ways to start and stop the Sybase Control Center in an Unwired Platform environment.

- By default, SybaseControlCenter.X.X is installed to run as a Windows service, and is set by the installer to start automatically.
- You can also use a command-line script as required.
- Start or stop from the Windows Control Panel; change automatic start and restart:
 - a) Open the Windows Control Panel.
 - b) Select **Administrative Tools > Services**.

- c) Locate SybaseControlCenter.X.X. If the service is running, the status column displays “Started.”
- d) To start or stop the service, right-click the service and choose **Start** or **Stop**.
- e) Double-click the service.
- f) To set the service to automatically start when the system starts, change the **Startup type** to Automatic.
- g) To restart the service in case of failover, choose the **Recovery** tab and change the First, Second, and Subsequent failures to Restart Service.
Click **Apply** to save the modifications before closing the dialog.
- Manually starting Sybase Control Center by command-line script:
 - a) Enter the start command:

```
<UnwiredPlatform_InstallDir>\SCC-X_X\bin\scc.bat
```
- Manually stopping Sybase Control Center by command-line script:
 - a) Enter the stop command:

```
<UnwiredPlatform_InstallDir>\SCC-X_X\bin\scc.bat -stop
```

Note: You can use **scc.bat -stop** only to stop an SCC that was manually started with "scc.bat"; it cannot stop the SCC windows service.

Cleaning Up the Flash Player Cache

Sybase recommends you clean up the Flash Player cache, after upgrading to the latest version of Sybase Control Center. This is needed if you have used a previous version of Sybase Unwired Platform on the same machine. This cleanup is only required once.

1. Navigate to `C:\Documents and Settings\username\Application Data\Macromedia\Flash Player\#SharedObjects`.
2. Delete all files under this folder.

Note: Alternatively, go to the following link from a browser: http://www.macromedia.com/support/documentation/en/flashplayer/help/settings_manager07.html. Use the Website Storage settings panel to change storage capacity, or delete Websites to clean up the cache.

Getting Started with Production Clusters

Get started using Sybase Control Center in production clusters of Unwired Platform. Follow steps to configure and prepare Sybase Control Center for Unwired Platform use.

Getting Started After Installing

Perform postinstallation testing and configuration.

Prerequisites

Start Sybase Control Center.

Task

1. Install Adobe Flash Player 10.1 or later in the Web browser you will use to connect to Sybase Control Center.

Flash Player is a free plug-in. You can download the latest version from <http://get.adobe.com/flashplayer/>.

If Flash Player is already installed but you are not sure which version you have, go to the Adobe test site at <http://adobe.com/shockwave/welcome>. Click the link that says **Test your Adobe Flash Player installation**. The version information box on the next page that appears displays your Flash Player version.

2. To connect to Sybase Control Center, direct your browser to:

```
https://<scc_server_hostname>:8283/scc
```

Note: If you changed the default HTTPS port during installation, use the new port number instead of 8283.

3. If you see an error about the security certificate, add Sybase Control Center to your browser's trusted sites zone (Internet Explorer) or add a security exception (Firefox).
4. Log in. Use the login account (supAdmin) and password that you set up during installation. This account can be used for both SCC login, and SUP login.
5. Learn about Sybase Control Center. To open the help system, click ? in the upper-right corner of the screen, or select **Help > Online Documentation**.

Setting Up Browser Certificates for Sybase Control Center Connections

To avoid security exceptions when launching Sybase Control Center, set up security certificates correctly.

This task is required when:

- The browser session starts from a host computer that is remote from the Sybase Control Center installation.
- The browser session starts on the same computer as Sybase Control Center and reports a Certificate Error. The installer automatically sets up a local security certificate, but the certificate installed for https in the web container keystore is a self-signed root certificate, which is not recognized by the client browser.
- The host computer does not have Visual Studio Certificate Manager SDK installed.

Alternatively, follow browser-specific instructions to accept the certificate into the Windows certificate store.

1. Change the default shortcut to use the full host name of the computer on which Sybase Control Center has been installed.

The host name is required because the default self-signed generated certificate the installer issues cannot be assigned to “localhost.”

For example, change the shortcut URL to something similar to:

```
"%ProgramFiles%\Internet Explorer\iexplore.exe" https://
SCCHost.mydomain.com:8283/scc
```

2. Add the certificate to the Windows certificates store.

- a) Extract the self-signed certificate:

```
<UnwiredPlatform_InstallDir>\JDKX.X.X.XX\bin\keytool.exe -
exportcert -alias jetty
-keystore <UnwiredPlatform_InstallDir>\SCC-X.X\services
\EmbeddedWebContainer\container\Jetty-X.X.XX\keystore -file
cert.crt
```

- b) Click **Start > Run**, type mmc, and then click **OK** to import the cert . crt file into the host computer’s Windows store with the Windows Certificate Manager. The default password for both the keystore and the alias is "changeit".

Logging Into Sybase Control Center with an Installer-Defined Password

The person acting as platform administrator logs in to Sybase Control Center for the first time after installation.

During installation, the person installing Unwired Platform defines a password for the supAdmin user. This password is used to configure the Preconfigured login module that performs the administrator authentication.

Note: This installer-defined password is not intended to be a permanent administrator credential. You must replace this module with a production-grade authentication module, typically LDAP.

1. Launch Sybase Control Center.
2. Enter supAdmin for the user name and type the <supAdminPwd> for the password.
3. Click **Login**.

Logging out of Sybase Control Center

Log out of a cluster when you finish your administration session.

In order to protect system security, Sybase recommends that you log out of Sybase Control Center when you are not using the console.

Choose one of these methods:

- Click the **Logout** link at the top right corner of the console.

Get Started

- From the Sybase Control Center menu, select **Application > Logout**.

Configuring Memory Usage

(Optional) Determine whether you need to configure how much memory Sybase Control Center uses, and if so which configuration method to use.

It is not usually necessary to configure memory usage for Sybase Control Center. This table lists memory options you can set and circumstances under which you should consider changing them.

Modify this value	When	Guidelines
<p>Maximum memory</p> <ul style="list-style-type: none">• <code>jvmopt=-Xmx</code> – if you are running SCC as a Windows service• <code>SCC_MEM_MAX</code> – if you are starting SCC from the command line	<ul style="list-style-type: none">• You need to prevent Sybase Control Center from using more than a given amount of memory• SCC fails to start and may display an error: <code>Could not create the Java Virtual machine.</code>• An <code>OutOfMemory</code> error says SCC is out of heap space• A warning message about system memory appears during the start process• The machine where SCC is installed has less than 2GB of memory. (Starting SCC on a machine with less than 2GB of memory triggers the startup warning message about system memory.)	<p>On machines with less than 2GB of memory, set maximum memory to 256MB or more.</p> <p>Default value: none. (On machines with 2GB or more of memory, maximum memory is set dynamically and is effectively limited only by the amount of system memory available.)</p>
<p>Permanent memory</p> <ul style="list-style-type: none">• <code>jvmopt=-XX:MaxPermSize</code> – if you are running SCC as a Windows service• <code>SCC_MEM_PERM</code> – if you are starting SCC from the command line	<p>An <code>OutOfMemory</code> error says SCC is out of permanent generation space</p>	<p>Increase by 32MB increments. If you reach a value equal to twice the default and still see the <code>OutOfMemory</code> error, contact Sybase technical support.</p> <p>Default value: 128MB</p>

You can change memory options in two ways:

- For Sybase Control Center started from the command line – execute commands to set one or more environment variables before executing the **scc** command to start Sybase Control Center. When you use this method, your changes to the memory options last only as long as the current login session. This method is useful for testing new option values.
- For the Sybase Control Center service – modify a file used by the SCC service. When you use this method, your changes to the memory options persist—Sybase Control Center uses them every time it starts as a service.

Changing a Memory Option on the Command Line

Before you start Sybase Control Center from the command line, you can issue a command to change the value of a memory option temporarily.

Changes made using this method last only as long as the current login session. This method is useful for testing new option values.

1. If Sybase Control Center is running, shut it down.
2. Set the environment variable. Specify a size in megabytes but do not indicate the units in the command.

```
> set SCC_MEM_MAX=512
```

3. Use the **scc** command to start Sybase Control Center.

Changing a Memory Option for an SCC Windows Service

Add a **jvmopt** command to the `scc.properties` file to change a memory option (`-Xmx` or `-XX:MaxPermSize`) for a Sybase Control Center Windows service.

When you use this method to set memory options, your changes are permanent—Sybase Control Center uses them every time it starts as a service.

1. If Sybase Control Center is running, shut it down.
2. Open the SCC properties file:


```
<SCC-install-directory>\SCC-3_2\bin\scc.properties
```
3. Add (or modify, if it already exists) a **jvmopt** line specifying the memory size in Java format. Use `m` for megabytes or `g` for gigabytes.

For example:

```
jvmopt=-Xmx512m
```

4. Save the file and start the Sybase Control Center Windows service.

Configuring the Automatic Logout Timer

(Optional) Set Sybase Control Center to end login sessions when users are inactive for too long.

Prerequisites

Launch Sybase Control Center and log in using an account with administrative privileges. (The login account or its group must have `sccAdminRole`.)

Task

1. From the menu bar, select **Application > Administration**.
2. Select **General Settings**.
3. Click the **Auto-Logout** tab.
4. Enter the number of minutes after which an idle user will be automatically logged out.
Enter 0 or leave the box empty to disable automatic logout.
5. Click **OK** (to apply the change and close the properties dialog) or **Apply** (to apply the change and leave the dialog open).

Manually Opening the Unwired Platform Console

If the Unwired Platform administration console does not appear automatically, you may need to manually open it in Sybase Control Center (SCC). Once open, you can then use the Unwired Platform administration console to manage the Unwired Server enabled mobile environment.

Prerequisites

Before managing a cluster, ensure that the login has SCC administration privileges.

Task

1. In the SCC menu, select **View > Open > Resource Explorer**.
2. From the list of resources, select the cluster you want to manage.
3. From the Resource Explorer menu bar, click **Resources > Add Resources to Perspective**.
The Unwired Server is added to the Perspective Resources window.
4. In the Perspective Resources window, mouse over the cluster you want to manage, click the down arrow, and select **Authenticate**.
5. To authenticate against the cluster, select one of these:
 - **Use my current SCC login** – SCC uses the administrator's initial SCC login credentials to establish a connection to the Unwired Platform cluster. Use this option if you have already mapped the SCC administrator role to the SUP administrator role.
 - **Specify different credentials** – enter a new user name and password specifically for logging in to this cluster. Use this option if SCC and Unwired Platform use different authentication repositories. Using different credentials in this step is unnecessary if SCC and Unwired Platform use the same security provider.
6. Click **OK**.
7. Mouse over the cluster you want to open, click the down arrow, and select **Manage**.

If you are successfully authenticated, the Unwired Platform console appears. If authentication fails, see *Sybase Control Center Issues* in the *Troubleshooting* guide.

Adding or Updating Unwired Server Registration Properties

By default a Sybase Control Center detects and registers clusters and Unwired Server nodes as managed resources of Sybase Control Center automatically: the resource entry named 'localhost' is created for the local server upon installation. However, you may need to manually register other new clusters or nodes or modify existing entries under specific conditions.

For information on these conditions, see *When Manual Managed Resource Property Changes Are Needed*.

1. Choose your action:

- To register a new resource, on the Sybase Control Center menu, select **Resource > Register**.
- To update the resource properties, on the Sybase Control Center menu, select **View > Select > Perspective Resources** view. Then in the Name column, click **EntryName > Properties**.

2. Configure any of these properties, depending on you initial action:

- the resource name and type
- a description
- host name and port of the server

The host name and port must match those configured for the Unwired Server management port.

3. If you changed hostname, reauthenticate the server:

- a) Click **EntryName > Clear Authentication** to remove currently validated credentials to the previous host values.
- b) Click **EntryName > Authenticate** to reauthenticate with the current host values.

4. Once authenticated, you can now manage it from Sybase Control Center: click **EntryName > Manage** to launch the Unwired Platform management console.

When Manual Managed Resource Property Changes Are Needed

Understand the conditions under which managed resource properties need to be manually edited or added

These are the conditions under which you must manually create a new registration entry:

- If a cluster or node is not located within your network.
- If it is not automatically detected and registered in the Sybase Control Center Resource Explorer

These are the conditions under which you must manually update an existing registration entry:

- If you modify the Unwired Server configuration to change the management port, you need to update these resource properties to match those values.

Note: When modifying the hostname of the resource, you need to reauthenticate the resource.

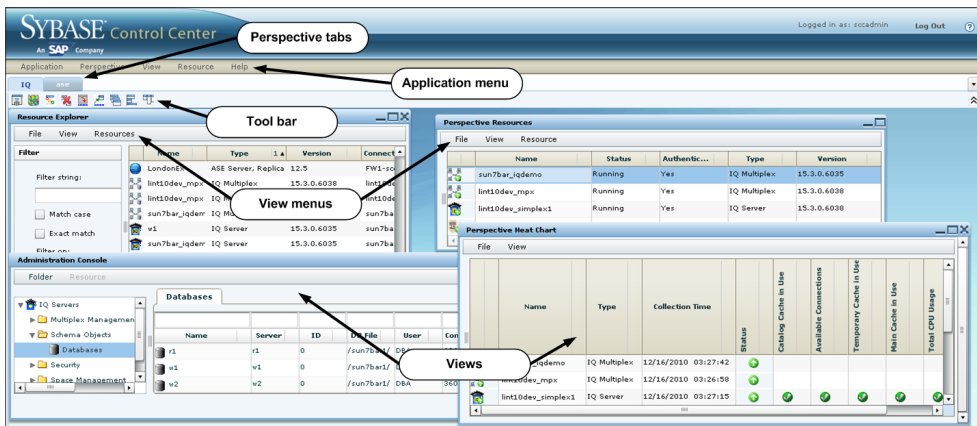
Understanding the Sybase Control Center Interface

Manipulate Sybase Control Center interface elements to set up the console according to your requirements and preference.

User Interface Overview

This illustration labels important elements of the Sybase Control Center user interface so you can identify them when they appear in other help topics.

Figure 1: Sybase Control Center User Interface










Toolbar Icons

Describes the icons in the Sybase Control Center toolbar for launching and managing views.

Table 7. Toolbar icons

Icon	Name	Description
	Show/Hide Perspective Resources View	Displays or minimizes the Perspective Resources view, which lists registered resources in this perspective.
	Launch Resource Explorer	Opens the resource explorer, which lists reachable resources (both registered and unregistered).

Icon	Name	Description
	Launch Heat Chart	Opens the perspective heat chart, which gives a status overview of the registered resources in this perspective.
	Close All Open Views	Closes all open and minimized views.
	Minimize All Views	Minimizes all open views.
	Restore All Minimized Views	Returns all minimized views to their original size.
	Cascade All Open Views	Arranges open views to overlap each other.
	Tile All Open Views Vertically	Arranges open views in a vertical manner.
	Tile All Open Views Horizontally	Arranges open views in a horizontal manner.

Sybase Control Center Functionality Not Applicable to Unwired Platform

Sybase Control Center is a standard management framework used by multiple products, including Sybase Unwired Platform. Certain standard functions that appear in the user interface cannot be used to administer Unwired Platform.

The following Sybase Control Center features can be disregarded in the context of Sybase Unwired Platform:

- Alerts
- Schedules
- Heat charts
- Historical performance monitoring
- Logging

These features either do not apply to Sybase Unwired Platform or are redundant due to custom functionality implemented in place of standard functions. The inapplicable Sybase Control Center functionality cannot be removed, as it may be required by other Sybase product servers also using Sybase Control Center.

Accessibility Features

This document is available in an HTML version that is specialized for accessibility. You can navigate the HTML with an adaptive technology such as a screen reader, or view it with a screen enlarger.

The Sybase CEP Option R4 documentation complies with U.S. government Section 508 Accessibility requirements. Documents that comply with Section 508 generally also meet

Get Started

non-U.S. accessibility guidelines, such as the World Wide Web Consortium (W3C) guidelines for Web sites.

For information about accessibility support in the Sybase IQ plug-in for Sybase Central™, see “Using accessibility features” in Chapter 1, “Introducing Sybase IQ” in Introduction to Sybase IQ. The online help for Sybase IQ, which you can navigate using a screen reader, also describes accessibility features, including Sybase Central keyboard shortcuts.

Note: You might need to configure your accessibility tool for optimal use. Some screen readers pronounce text based on its case; for example, they pronounce ALL UPPERCASE TEXT as initials, and MixedCase Text as words. You might find it helpful to configure your tool to announce syntax conventions. Consult the documentation for your tool.

For information about how Sybase supports accessibility, see Sybase Accessibility at <http://www.sybase.com/accessibility>. The Sybase Accessibility site includes links to information on Section 508 and W3C standards.

Sybase Control Center Accessibility Information

Sybase Control Center uses the Adobe Flex application.

For the most current information about Adobe Flex keyboard shortcuts, see http://livedocs.adobe.com/flex/3/html/help.html?content=accessible_5.html.

Note: To use Sybase Control Center with JAWS for Windows screen reading software effectively, download and install the appropriate Adobe scripts. See www.adobe.com.

Perspectives

A perspective is a named container for a set of one or more managed resources. You can customize perspectives to provide the information you need about your environment.

As the main workspaces in the Sybase Control Center window, perspectives let you organize managed resources. You might assign resources to perspectives based on where the resources are located (continents, states, or time zones, for example), what they are used for, which group owns them, or which administrator manages them. Perspectives appear as tabs in the main window.

Every perspective includes a Perspective Resources view, which lists the resources in that perspective and provides high-level status and descriptive information. Use the View menu to switch from detail view to icon view and back.

You can open additional views as needed to manage the perspective’s resources. The views in a perspective display information only about resources in that perspective.

One resource can appear in many perspectives.

Creating a Perspective

Create a perspective in which you can add and manage resources.

1. From the application menu bar, select **Perspective > Create**.
2. Enter a name for your perspective. The name can contain up to 255 characters.
3. Click **OK**.

Removing a Perspective

Delete a perspective window.

1. Select the perspective tab you want to delete.
2. In the main menu bar, select **Perspective > Delete**.
The selected perspective disappears. If there are other perspectives, Sybase Control Center displays one.

Renaming a Perspective

Change the name of your perspective.

1. Select the perspective tab you want to rename.
2. From the main menu bar, select **Perspective > Rename..**
3. Enter the new name for your perspective.
4. Click **OK**.

Views

Use views to manage one or more resources within a perspective.

In Sybase Control Center, views are the windows you use to monitor and manage a perspective's resources. You can re-arrange, tile, cascade, minimize, maximize, and generally control the display of the views in your perspective.

Each perspective includes these views:

- Perspective Resources
- Administration Console

Managing a View

Open, close, minimize, maximize, or restore a view in the current perspective.

You can:

Task	Action
Open a view	Do one of the following: <ul style="list-style-type: none"> • In the Perspective Resources view, click a resource, pull down its menu using the handle to the right of the resource name, and select the view to open. • In the application menu bar, select View > Open and choose a view.
Close a view	Select the view to close. In the application menu bar, select View > Close . You can also click the X in the view's upper right corner.
Maximize a view	Click the box in the view's upper right corner. The view enlarges to fill the entire perspective window. Click the box again to return the view to its former size.
Minimize a view	Click the _ in the view's upper right corner. The view shrinks to a small tab at the bottom of the perspective window.
Minimize all views	In the application menu bar, select View > Minimize All Views .
Restore a view	Click the box on the minimized tab to maximize the view. Click the box again to return the view to its former (smaller) size so you can see other views at the same time.
Bring a view to the front	In the application menu bar, select View > Select and choose the view you want from the submenu.

Arranging View Layout in a Perspective

Use the view layout options to manage your perspective space.

Click one of these icons from the Sybase Control Center toolbar:

- **Cascade all open views**
- **Tile all open views vertically**
- **Tile all open views horizontally**

In a cascade, views overlap; in tiling arrangements, they do not.

Alternatively, you can arrange view layouts from the Sybase Control Center menu bar. From the menu bar, select **Perspective > Arrange** and select your view layout.

Repository

The Sybase Control Center embedded repository stores information related to managed resources, as well as user preference data, operational data, and statistics.

You can back up the repository database on demand, schedule automatic backups, restore the repository from backups, and configure repository purging options. Full and incremental

backups are available. A full backup copies the entire repository. An incremental backup copies the transaction log, capturing any changes since the last full or incremental backup.

By default, Sybase Control Center saves backups as follows:

- Each full backup is stored in its own subdirectory in <SCC-install-directory>/backup.
- Each incremental backup is stored in a file in <SCC-install-directory>/backup/incremental.

Sybase recommends that you periodically move backup files to a secondary storage location to prevent the installation directory from becoming too large.

Scheduling Backups of the Repository

Configure full and incremental backups of the repository to occur automatically.

Prerequisites

Determine your backup strategy, including when to perform full backups and incremental backups. For example, you might schedule incremental backups every day and a full backup every Saturday.

You must have administrative privileges (sccAdminRole) to perform this task.

Task

A full backup copies the entire repository. An incremental backup copies the transaction log, capturing any changes since the last full or incremental backup.

1. From the main menu, select **Application > Administration**.
2. In the left pane, select **Repository**.
3. Click the **Full Backup** tab.
4. (Optional) To change the directory in which backups will be stored, click **Browse** and navigate to the desired directory.
5. Select **Schedule a Regular Backup**.
6. Specify the day you want scheduled backups to begin. Enter a **Start date** or click the calendar and select a date.
7. (Optional) Use the **Time** and **AM/PM** controls to specify the time at which backups occur.
8. Specify how often backups occur by setting the **Repeat interval** and selecting hours, days, or weeks.
9. (Optional) To purge the repository after each backup, select **Run a repository purge after the backup completes**.
10. If you include purging in the backup schedule, go to the **Size Management** tab and unselect **Automatically purge the repository periodically** to disable automatic purging.

Get Started

11. Click **Apply** to save the schedule.
12. Click the **Incremental Backup** tab and repeat the steps above to schedule incremental backups to occur between full backups.

Next

Set purging options on the Size Management tab.

Modifying the Backup Schedule

Suspend or resume repository backups or change the backup schedule.

Prerequisites

You must have administrative privileges (sccAdminRole) to perform this task.

Task

1. From the main menu, select **Application > Administration**.
2. In the left pane, select **Repository**.
3. Choose the type of backup to modify:
 - Click the **Full Backup** tab, or
 - Click the **Incremental Backup** tab.
4. (Optional) To suspend or resume the backup schedule, select or unselect **Schedule a Regular Backup**.
When you unselect (uncheck) this option, the scheduling area is grayed out and scheduled backups no longer occur. However, the schedule is preserved and you can reinstate it at any time.
5. To change the backup schedule, edit the **Start date**, **Time**, **Repeat interval**, or units. You can also select or unselect **Run a repository purge after the backup completes**.
6. Click **Apply** to save the schedule.

Forcing an Immediate Backup

Perform an unscheduled full or incremental backup of the repository.

Prerequisites

You must have administrative privileges (sccAdminRole) to perform this task.

Task

1. From the main menu, select **Application > Administration**.
2. In the left pane, select **Repository**.
3. Choose the type of backup to run:

- Click the **Full Backup** tab, or
- Click the **Incremental Backup** tab.

4. Click **Back up Now**.

Sybase Control Center saves the backup to the directory shown in the Location field.

Restoring the Repository from Backups

Load backup files into the repository database to revert undesirable changes or to recover from a catastrophic failure.

If you configured Sybase Control Center to store backups somewhere other than the default location, change the source directory in the copy commands in this procedure.

1. Shut down Sybase Control Center.
2. Copy the most recent full backup from `<SCC-install-directory>/backup/<generated_directory_name>` to `<SCC-install-directory>/services/Repository`. For example:

```
copy C:\sybase\SCC-3_2\backup\repository.
270110161105\scc_repository.db
C:\sybase\SCC-3_2\services\Repository
```

3. If you have no incremental backups to load,
 - a) Also copy the log file from `<SCC-install-directory>/backup/<generated_directory_name>` to `<SCC-install-directory>/services/Repository`. For example:

```
copy C:\sybase\SCC-3_2\backup\repository.
270110161105\scc_repository.log
C:\sybase\SCC-3_2\services\Repository
```

- b) Skip to step 5 on page 25.

4. Start the repository database using the **-ad** option, which directs it to load transaction logs (incremental backups) from the `incremental` directory. (The database loads full backups automatically.) For example:

```
cd <SCC-install-directory>\services\Repository

..\..\bin\sa\bin_<platform>\dbsrv11.exe scc_repository -ad
<SCC-install-directory>\backup\incremental
```

The repository database loads the full backup and any subsequent incremental backups present in the `incremental` directory. Incremental backups are loaded in date order. After loading and saving, the database shuts down.

5. Start Sybase Control Center.

If you loaded incremental backups, SCC starts normally (that is, no further recovery occurs). If you copied a full backup to the `Repository` directory, the database recovers the repository from the full backup.

Example: Loading incremental backups into the repository database

These commands start SQL Anywhere® on a 32-bit Windows machine:

```
% cd C:\sybase\SCC-3_2\services\Repository  
  
% ..\..\bin\sa\bin_windows32\dbsrv11.exe scc_repository -ad  
C:\sybase\SCC-3_2\backup\incremental
```

Configuring Repository Purging

Change repository purging options.

Prerequisites

You must have administrative privileges (sccAdminRole) to perform this task.

Task

As you decide how to purge your repository, consider that:

- Purging keeps the repository from absorbing too much disk space.
- By default, purging is enabled. It occurs once a day and purges data older than one day.
- Statistics and alert history can help you detect trends in server performance and user behavior. The Sybase Control Center statistics chart can graph performance data over a period of a year or more if the data is available. If you have enough disk space, consider saving data for a longer period of time or disabling the purging of statistics or alert history.
- Changing the purge frequency and other options might affect Sybase Control Center performance.

Note: If you configure purging as part of a scheduled backup of the repository, disable automatic purging on the Size Management tab.

1. From the main menu bar, select **Application > Administration**.
2. Select **Repository**.
3. Click the **Size Management** tab.
4. To turn automatic purging on or off, click **Automatically purge the repository periodically**.
Turn this option off if purging is configured as part of your scheduled full or incremental backups.
5. Click purge options to turn them on or off:
 - **Purge statistics**
 - **Purge alert history**
6. In **Purge data older than**, enter the number of days after which to purge repository data.
7. Click **Apply**, then **OK**.

Sybase Control Center Console

The console is a command-line interface for displaying details about the status of the Sybase Control Center server and its subsystems.

When you use the **scc** command to start Sybase Control Center, it displays start-up messages and then displays the console prompt.

Note: The console prompt does not appear if you start Sybase Control Center as a service, if you direct the output of **scc** to a file, or if you start Sybase Control Center in the background.

Console Commands

Use the Sybase Control Center console to get status information on Sybase Control Center and its ports, plug-ins, and services.

help Command

Display syntax information for one or more Sybase Control Center console commands.

Syntax

```
help [command_name]
```

Parameters

- **command_name** – optional. status, info, or shutdown. If you omit *command_name*, **help** returns information on all the console commands.

Examples

- **Example 1** – returns information on the **status** command:

```
help status
```

Permissions

help permission defaults to all users. No permission is required to use it.

info Command

Display information about specified parts of the Sybase Control Center server.

If you enter **info** with no parameters, it returns information for every parameter.

Syntax

```
info [-a | --sys]
[-D | --sysprop [system-property]]
[-e | --env [environment-variable]]
[-h | --help]
[-m | --mem]
```

```
[ -p | --ports ]  
[ -s | --services ]
```

Parameters

- **-a | --sys** – optional. List all the services known to Sybase Control Center, indicate whether each service is enabled, and list other services on which each service depends.
- **-D | --sysprop [system-property]** – optional. Display information about the specified Java system property. Omit the system-property argument to return a list of all Java system properties and their values.
- **-e | --env [environment-variable]** – optional. List all the environment variables in the Sybase Control Center Java VM process environment. Omit the environment-variable argument to return a list of environment variables and their values.
- **-h | --help** – optional. Display information about the **info** command.
- **-m | --mem** – optional. Display information about the server's memory resources.
- **-p | --ports** – optional. List all the ports on which the Sybase Control Center agent and its services listen, indicate whether each port is in use, and show the service running on each port.
- **-s | --services** – optional. List all Sybase Control Center services, indicate whether each service is enabled, and list other services on which each service depends.

Examples

- **Example 1** – displays information about ports on this Sybase Control Center server:

```
info -p
```

Permissions

info permission defaults to all users. No permission is required to use it.

shutdown command

Stop the Sybase Control Center server if it is running.

Syntax

```
shutdown
```

Examples

- **Example 1** – shuts down Sybase Control Center:

```
shutdown
```

Permissions

shutdown permission defaults to all users. No permission is required to use it.

status Command

Display the status of the SCC agent, plug-in, or service components of Sybase Control Center.

Syntax

```
status [-a | --agent]
[-h | --help]
[-p | --plugin [plugin-name]]
[-s | --service [service-name]]
```

Parameters

- **-a | --agent** – display the status of the Sybase Control Center agent component.
- **-h | --help** – display information about the **info** command.
- **-p | --plugin [plugin-name]** – display the status of the specified Sybase Control Center plug-in (for example, ASEMap, the Adaptive Server® management module). Omit the plugin-name argument to return a list of plug-ins.
- **-s | --service [service-name]** – display the status of the specified Sybase Control Center service (for example, the Alert service or the Messaging service). Omit the service-name argument to return a list of services.

Examples

- **Example 1** – displays status information on the Repository service:

```
status --service Repository
```

Permissions

status permission defaults to all users. No permission is required to use it.

Sybase Control Center Security

User access to Sybase Control Center is controlled by configuring a security provider. Security providers are configured with the Unwired Platform management console.

By default, Sybase Control Center delegates user access control to providers configured for Unwired Server. Consequently, the login and group management features for Sybase Control Center (that is, those available when you click **Application > Administration > Security** from the Sybase Control Center menu) do not apply to the Unwired Platform use case. See *Securing Platform Administration* in the *Security* guide.

Administer

Use Sybase Control Center for Unwired Platform to administer and configure components of a cluster registered as a managed resource. When you configure cluster components you are setting up the elements required to mobilize your data. Once configured you perform ongoing administration tasks to maintain the environment.

Clusters

As an organization grows, Unwired Platform administrators need to create a scalable IT infrastructure using clusters. Clustering creates redundant Unwired Platform components on your network to provide a highly scalable and available system architecture.

Organizations can seamlessly achieve high availability and scalability by adding more or redundant instances of core components. Redundant instances of critical components provide transparent failover.

In a production environment, the Unwired Platform deployment typically uses at least one relay server. The connections to relay servers can be configured within a cluster instance from Sybase Control Center.

Cluster-Affecting Configuration Changes

Before you configure Unwired Servers in a cluster, ensure you understand how changes are synchronized to cluster members.

When you make a cluster-affecting change on the primary Unwired Server, those changes are synchronized to all secondary servers in the cluster. This ensures that servers are configured the same way and behave consistently within the cluster.

Cluster-affecting changes include:

- server configuration
- monitoring setup
- security configuration

Copying and Pasting Properties

Values displayed in property tables in Sybase Control Center can be copied and pasted.

Tables that support copying and pasting include monitoring properties, device properties, user properties, registration templates, domain log properties, and sever log properties.

1. To copy a value, right click the cell, then select **Copy** from the context menu.
2. To paste what you have copied, go to the property table you require, click the cell in question, then select **Paste** from the context menu. You cannot paste in a table cell that is read only, by you can copy a value from a table cell and paste it elsewhere (for example, copy text input for a search).

Configuring General Cluster Properties

Configure properties of a cluster to control whether or not a shared data folder is used, or whether asynchronous operation replays are enabled for all cluster packages.

1. In Sybase Control Center navigation pane, click the name of the cluster.
2. In the administration view, click **General**.
3. To synchronize data across Unwired Server nodes in the cluster:

In a production environment, Sybase recommends using a shared data folder to manage communications between the primary node in the cluster, and secondary nodes. For details, see *Shared Data Folders* in *System Administration*.

- a) To enable cluster synchronization, click **Cluster sync data shared path enabled**.
- b) Configure the shared data path.

In **Cluster sync shared data path** set the shared folder using the UNC convention (`\\hostname\shared_folder_name`). The shared folder must have write access.

4. Configure the queue limit for asynchronous operation replays in **Asynchronous operation replay queue count**. The minimum acceptable queue count is 1 and the default is 5.

The queue for operation replays can then be viewed by a package. See *Viewing Asynchronous Operation Replays*.

Viewing Cluster Information

View cluster information to determine the name and size of the cluster.

1. In Sybase Control Center navigation pane, click the name of the cluster.
2. Review information for general properties:
 - The name of the cluster. By default the cluster name is the name of the host computer upon which the primary Unwired Server node was installed with a `_cluster` suffix appended to the host name.
 - The number of servers and outbound enablers that are members of the cluster.
 - The cluster sync data shared path, if enabled.
 - The asynchronous operation replay queue count. See *Configuring General Cluster Properties*.

Checking System Licensing Information

Review licensing information to monitor available and used device licenses, license expiry dates, and other license details. This information allows administrators to manage license use and determine whether old or unused device licenses should be transferred to new devices.

1. In the left navigation pane, select the top-level tree node.
2. In the right administration pane, select the **General** tab, and click **Licensing**.
3. Review the following licensing information:
 - Server license type – the type of license currently used by Unwired Platform. For more information on license types, see *Sybase Unwired Platform Licenses* in *System Administration*.
 - Production edition – the edition of the software you have installed.
 - Server license expiry date – the date and time at which the server license expires. When a server license expires, Unwired Server generates a license expired error and Unwired Server is stopped.
 - Overdraft mode – allows you to generate additional licenses in excess of the quantity of licenses you actually purchased. This enables you to exceed your purchased quantity of licenses in a peak usage period without impacting your operation. This mode is either enabled or disabled, as specified by the terms of the agreement presented when you obtain such a license.
 - Total device license count – the total number of device licenses available with your license. This count limits how many devices can connect to your servers. See *Sybase Unwired Platform Licenses* topics in *System Administration* for licensing information.
 - Used device license count – the total number of unique devices associated with the users currently registered with the server. If all of your available device licenses are in use, you can either upgrade your license or manually delete unused devices to make room for new users:
 - For workflow and Online Data Proxy client devices, delete the Application Connections that are no longer in use.
 - For native replication-based applications, delete Package Users on the respective Package.
 - For native messaging-based applications, delete the Application Connections associated with the clients not in use.See *System Administration* for licensing information.
 - Device license expiry date – the date and time at which the device license expires. When a device license expires, Unwired Server generates a license expired error and connection requests from registered devices are unsuccessful.
 - Used mobile user license count – the number of mobile user licenses currently in use. A mobile user is a distinct user identity—username and associated security configuration

—that is registered in the server. As such, the used mobile user license count represents the total distinct user identities registered on the server. One mobile user may access:

- Multiple applications and different versions of the same application.
- The same or different versions of an application from multiple devices.
- Used application user license count – the number of all registered application users of all applications. This value represents the cumulative total of the distinct user identities registered for each application. The same user identity using:
 - Multiple versions of the same application counts as one application user.
 - Two different applications count as two application users.

4. Click **Close**.

Note: Unwired Platform licensing is configured during installation. However, if necessary, license details can be changed at a later time. See *Manually Updating and Upgrading License Files* in *System Administration*.

Checking Cluster Status

Verify that a cluster is running.

In the left navigation pane, check the status (in brackets) beside the cluster name.

Relay Server

Relay Server acts as a reverse proxy for client devices communicating with the Unwired Server cluster, and it provides load balancing for the Unwired Server cluster.

Relay Servers are deployed on the DMZ subnet. With a corresponding Outbound Enabler (RSOE), Relay Server enables communication from the Unwired Server cluster to client devices, via the Internet, without opening an inbound port on the internal firewall.

Each Unwired Server instance is supported by one or more RSOEs. Each RSOE opens outbound connections to the Relay Server, to handle both inbound and outbound communication channels, on behalf of the Unwired Server. Connections between the RSOE and Relay Server use HTTPS protocol.

Relay Server also provides load balancing for the Unwired Server cluster by forwarding requests from client devices to Unwired Servers in the cluster, by round-robin distribution.

You must configure Unwired Server to use Relay Server, using these high-level steps::

1. Use Sybase Control Center to configure a Unwired Server cluster with farms, and nodes and their tokens, as needed, and with relay server connection information.
2. Generate the relay server configuration file from Sybase Control Center, and use it to update the relay server configuration (manually transfer the generated file to the relay server node and use **rshost.exe** utility to update the configuration). Refer to *System*

Administration or Installation Guide for Runtime for relay server installation and configuration information.

3. Set up Outbound Enablers on each Unwired Server node.

Configuring Unwired Server to use Relay Server

Choose a method for configuring Unwired Server to use Relay Server, then generate a Relay Server configuration file. Copy the file to the Relay Server host, and quickly distribute the same configuration to multiple Relay Server nodes, with minimal changes.

This task applies only to a Relay Server installed on the LAN. It does not apply to the Sybase Hosted Relay Service.

Configuring Relay Server Properties

There are two methods of configuring Relay Server properties.

Choose from one of these methods. Once completed, transfer the resulting configuration file to all hosts upon which Relay Server has been installed. For installation details, see *Installing a Relay Server* in *Installation Guide for Runtime*.

Creating a Quick Configuration

Create a Relay Server configuration primarily with system defaults, and create Outbound Enabler (RSOE) processes for each Unwired Server.

1. In the navigation pane, click the Unwired Server cluster name.
2. In the administration pane, click the **Relay Servers** tab.
3. Click **Quick Configure**.
4. Specify these property values:
 - **Host** – either the host name of the Relay Server, or the host name of the load balancer (if any).
 - **Http port** – the Relay Server HTTP port.
 - **Https port** – the Relay Server HTTPS port.
 - **URL suffix** – the URL suffix used by the Outbound Enabler to connect to a Relay Server.
 - **Replication or Messaging farm token** – the security token used by the Outbound Enabler to authenticate its connection with the Relay Server. Assign a token string (up to 2048 characters); this same token can be shared by replication or messaging farms. The replication and messaging farm token values can be the same.
 - **(Optional) Description** – a user-definable description of the Relay Server.
5. (Optional) Check **Advanced settings** and specify these property values:
 - **Http user name** – user name for RSOE authentication on the Web server (Relay Server host).
 - **Http password** – password for RSOE authentication on the Web server.

6. (Optional) For Online Data Proxy deployments, configure connection values to required Internet proxy servers:
 - **Proxy server host** – host name of the Internet proxy server.
 - **Proxy server port** – connection port on the Internet proxy server.
 - **Http proxy user** – user name for RSOE authentication on the Internet proxy server.
 - **Http proxy password** – password for RSOE authentication on the Internet proxy server.
7. Click **OK** to generate a Relay Server configuration file, and the RSOE processes for each Unwired Server.

Properties in the [backend_farm] and [backend_server] sections are populated automatically, based on the Unwired Server's cluster name and host name.

Six RSOE instances (three each for messaging and replication ports) are created on each Unwired Server host, but they not started.

Next

Review the values in the Relay Server configuration file, and edit if necessary.

Creating a Custom Relay Server Configuration

Create a Relay Server configuration by specifying all configuration property values.

Launching the Relay Server Configuration Wizard

Launch the Relay Server Configuration wizard to create a configuration file with customized property values.

1. In the navigation pane, click the Unwired Server cluster name.
2. In the administration pane, click the **Relay Servers** tab.
3. Click **New**.

Setting Relay Server General Properties

Set basic connection properties for the Relay Server.

Prerequisites

Launch the Relay Server configuration wizard.

Task

1. Specify these property values:
 - **Host** – either the host name of the Relay Server, or the host name of the load balancer (if any).
 - **Http port** – the Relay Server HTTP port.

- **Https port** –

Note: If Relay Server uses HTTPS and certificates, clients other than replication may not be able to connect: messaging applications only support HTTP, and hybrid workflow container applications for iOS support HTTPS — but not certificates.

the Relay Server HTTPS port.

- **URL suffix** – the URL suffix used by the Outbound Enabler to connect to a Relay Server.
 - **(Optional) Description** – a user-definable description of the Relay Server.
2. Add or remove HTTP credentials as required:
 - a) Check **Configure relay server HTTP credentials**.
 - b) To add new credentials, specify these property values and click +:
 - **User name** – user name for RSOE authentication on the Web server (Relay Server host).
 - **Password** – password for RSOE authentication on the Web server.
 - c) To remove credentials from the list, select the corresponding user name, then click **X**.
 3. Click **Next**.

Define Server Farms and Cluster Nodes

Set connection properties for the Unwired Server cluster and its constituent nodes.

Repeat these steps to add or remove multiple Unwired Server clusters, as needed.

1. Define the Unwired Server cluster.
 - a) Specify these property values:
 - **Farm ID** – a string that identifies the Unwired Server cluster, for which the Relay Server manages requests. This property is case-sensitive, and it must match the value in the Outbound Enabler configuration.
 - **Type** – the type of request managed by the Relay Server, Replication or Messaging protocol.
 - **(Optional) Description** – user-definable description of the Unwired Server cluster.
 - b) Click +.
 - c) Repeat steps 1 and 2 to add multiple Unwired Server clusters.
 - d) To delete a configured Unwired Server cluster, select it in the list, then click the **X** button.
2. Identify each Unwired Server instance in the cluster.
 - a) Select an existing Unwired Server cluster.
 - b) Specify these property values:

Administer

- **Node ID** – a string that identifies the Unwired Server in the cluster. This property is case-sensitive, and it must match the value in the RSOE configuration.
 - **Token** – the security token used by the Outbound Enabler to authenticate its connection with the Relay Server. Assign a token string (up to 2048 characters); this same token can be shared by replication or messaging farms.
- c) Click +.
 - d) Repeat steps 1 and 2 to add Unwired Server cluster nodes.
 - e) To delete a configured Unwired Server node, select it in the list and click **X**.
3. Click **Next** to review your settings, or click **Finish** to exit the wizard.

Note: After each change, you may need to update the relay server configuration, and take steps to manually update the relay server configuration.

Reviewing Configured Relay Server Properties

Review the Relay Server configuration to confirm property values, before you generate the configuration file.

1. Confirm the property values, ensuring that:
 - No errors exist.
 - All Unwired Server clusters are defined, and the correct type.
2. Click **Finish**.

The Relay Server is registered with Sybase Control Center, and it can be managed from the **Relay Servers** tab for the Unwired Server cluster.

Next

When you have finished adding all required Relay Servers, set up one or more Outbound Enabler (RSOE) for each Unwired Server in the cluster.

Generating the Relay Server Outbound Enabler Configuration File

To quickly and easily replicate a common Outbound Enabler (RSOE) configuration to multiple hosts, generate an RSOE configuration file.

Administrators can use Sybase Control Center to configure an initial RSOE for development. Once a configuration proves valid and stable, the administrator can generate the RSOE configuration file, then use `regRelayServer.bat` to apply it to Unwired Server hosts.

1. In the navigation pane, click the Unwired Server cluster name.
2. In the administration pane, click the **Relay Servers** tab.
3. Click **Generate**.
4. Choose **Relay server outbound enabler configuration XML file**, then click **Next**.
5. Select an output target for the file.

6. Click **Finish**.

Generating the Relay Server Configuration File

Generate all or part of a Relay Server configuration file. Then transfer the generated file to all Relay Server hosts.

Generating a configuration file extracts the property values stored in the cluster database during the configuration process, and writes them to a file.

1. In the navigation pane, click the Unwired Server cluster name.
2. In the administration pane, click the **Relay Servers** tab.
3. Click **Generate**.
4. Choose **Relay server properties configuration file**.
5. Select the parts of the file to generate:
 - The entire Relay Server configuration
 - A server node definition
 - A farm definition
6. Select an output target for the file.
7. Click **Finish**.

Managing Configured Relay Servers

Relay Servers configured with Sybase Control Center are registered in the Unwired Server cluster database. Administrators can view or edit configuration properties, and delete Relay Servers in Sybase Control Center when they are displayed in the **Relay Server** tab.

Viewing or Editing Relay Server Properties

View or edit configuration properties for a selected Relay Server.

Relaunching the Relay Server Configuration Wizard

Relaunch the Relay Server Configuration wizard to create a new Relay Server configuration file, with customized property values.

1. In the navigation pane, click the Unwired Server cluster name.
2. In the administration pane, click the **Relay Server** tab.
3. Select a Relay Server.
4. Click **Properties**.

Setting Relay Server General Properties

Set basic connection properties for the Relay Server.

Prerequisites

Launch the Relay Server configuration wizard.

Task

1. Specify these property values:

- **Host** – either the host name of the Relay Server, or the host name of the load balancer (if any).
- **Http port** – the Relay Server HTTP port.
- **Https port** –

Note: If Relay Server uses HTTPS and certificates, clients other than replication may not be able to connect: messaging applications only support HTTP, and hybrid workflow container applications for iOS support HTTPS — but not certificates.

the Relay Server HTTPS port.

- **URL suffix** – the URL suffix used by the Outbound Enabler to connect to a Relay Server.
 - **(Optional) Description** – a user-definable description of the Relay Server.
- ### 2. Add or remove HTTP credentials as required:
- a) Check **Configure relay server HTTP credentials**.
 - b) To add new credentials, specify these property values and click +:
 - **User name** – user name for RSOE authentication on the Web server (Relay Server host).
 - **Password** – password for RSOE authentication on the Web server.
 - c) To remove credentials from the list, select the corresponding user name, then click **X**.
- ### 3. Click Next.

Define Server Farms and Cluster Nodes

Set connection properties for the Unwired Server cluster and its constituent nodes.

Repeat these steps to add or remove multiple Unwired Server clusters, as needed.

1. Define the Unwired Server cluster.

- a) Specify these property values:
 - **Farm ID** – a string that identifies the Unwired Server cluster, for which the Relay Server manages requests. This property is case-sensitive, and it must match the value in the Outbound Enabler configuration.
 - **Type** – the type of request managed by the Relay Server, Replication or Messaging protocol.
 - **(Optional) Description** – user-definable description of the Unwired Server cluster.
- b) Click +.
- c) Repeat steps 1 and 2 to add multiple Unwired Server clusters.

- d) To delete a configured Unwired Server cluster, select it in the list, then click the **X** button.
2. Identify each Unwired Server instance in the cluster.
 - a) Select an existing Unwired Server cluster.
 - b) Specify these property values:
 - **Node ID** – a string that identifies the Unwired Server in the cluster. This property is case-sensitive, and it must match the value in the RSOE configuration.
 - **Token** – the security token used by the Outbound Enabler to authenticate its connection with the Relay Server. Assign a token string (up to 2048 characters); this same token can be shared by replication or messaging farms.
 - c) Click +.
 - d) Repeat steps 1 and 2 to add Unwired Server cluster nodes.
 - e) To delete a configured Unwired Server node, select it in the list and click **X**.
3. Click **Next** to review your settings, or click **Finish** to exit the wizard.

Note: After each change, you may need to update the relay server configuration, and take steps to manually update the relay server configuration.

Reviewing Configured Relay Server Properties

Review the Relay Server configuration to confirm property values, before you generate the configuration file.

1. Confirm the property values, ensuring that:
 - No errors exist.
 - All Unwired Server clusters are defined, and the correct type.
2. Click **Finish**.

The Relay Server is registered with Sybase Control Center, and it can be managed from the **Relay Servers** tab for the Unwired Server cluster.

Next

When you have finished adding all required Relay Servers, set up one or more Outbound Enabler (RSOE) for each Unwired Server in the cluster.

Deleting a Relay Server Configuration

Delete a Relay Server configuration to remove all defined Unwired Server clusters, server nodes, and RSOEs that connect to the Relay Server.

1. In the navigation pane, click the Unwired Server cluster name.
2. In the administration pane, click the **Relay Server** tab.
3. Select a Relay Server.

4. Click **Delete**.

Refreshing the Relay Server List

Refresh the Relay Server list to display current information about deployed and configured Relay Servers.

1. In the navigation pane, click the Unwired Server cluster name.
2. In the administration pane, click the **Relay Server** tab.
3. Select a Relay Server.
4. Click **Refresh**.

Relay Server Tab Reference

Configuration property values that appear in the Relay Server tab for an Unwired Platform cluster.

Column	Description
Host	either the host name of the Relay Server, or the host name of the load balancer (if any).
Http port	the Relay Server HTTP port.
Https port	the Relay Server HTTPS port.
URL suffix	the URL suffix used by the Outbound Enabler to connect to a Relay Server.

Unwired Server

The Unwired Platform runtime server is called Unwired Server. Unwired Server manages the data exchange process between the enterprise and device clients to create a homogeneous layer in a diverse mobile ecosystem. In a production environment, the Unwired Server must be installed on a 64-bit host.

Unwired Server features include:

- Data services – supports connections to back-end data resources using these standard technologies: enterprise databases with JDBC™ connections and Web Services (SOAP-style and REST-style) . Also supports connections to enterprise applications such as SAP®.
- Data virtualization – introduces a layer called a mobile business object (MBO) between your enterprise databases or applications, and the remote database on the device client. Utilizes a cache database (CDB) to optimize device client access and minimize back-end resource utilization.

- Device connection services – supports connections from various different platforms and operating systems with different communication styles.
 - Replication-based synchronization – A synchronization method where cached data is downloaded to and uploaded from client database to server via replication. Typically, mobile replication-based synchronization is used in occasionally connected scenarios.
 - Messaging-based synchronization – In flight messages are queued in a messaging cache. Synchronization occurs as messages are delivered to the device. Typically, mobile messaging-based synchronization is used in always available and occasionally disconnected scenarios.

Server List

Depending on the license you purchase and the type of environment you install, you may deploy multiple Unwired Servers in a cluster.

If you have installed multiple servers as part of a clustered architecture, you must register these servers first. Only servers that are installed on the same host as Sybase Control Center are registered automatically. Once registered, remote servers also appear in the server list.

Servers are listed according to their cluster mode (that is, primary or secondary servers). Sybase Control Center automatically identifies the primary server and lists it first, followed by secondary servers.

Stopping and Starting a Server

Stop and start a server to perform maintenance or to apply changes to server settings. You can perform this action as a two-step process (stop and start) or as a single restart process.

You can stop and start a server from Sybase Control Center for servers that are installed on the same host as Sybase Control Center, as well as servers that are installed on different hosts.

Note: If someone manually shuts the server down, this action triggers multiple errors in Sybase Control Center for Unwired Server until the console determines that the server is no longer available. This takes approximately 30 seconds to detect. When this occurs you might see multiple `Runtime API throws exception` errors logged. Wait for the server to come online and log into the server again to resume your administration work.

1. In the Sybase Control Center navigation pane, click **Servers** to display the servers list.
2. Select one or more servers in this list.
3. Choose an appropriate process:
 - To stop the server, click **Stop**. You can then perform the administration actions you require that might require the server to be started. To then restart the server, click **Start**.
 - If you perform an administration action that requires a restart to take effect, click **Restart**. This shuts the server down and restarts it in a single process.

As the server stops and starts, progress messages display in the Server Console pane.

Starting and Stopping RSOE

Start and stop an RSOE process as needed. All configured RSOEs are started by default when the Unwired Server starts.

1. In the navigation pane, click **Servers > ServerNode > Server Configuration**.
2. In the administration pane, select the Outbound Enabler tab, select the RSOE instances, and click **Start** or **Stop**.

Suspending and Resuming a Server

Suspend and resume a server to temporarily disallow clients to access the specific server for routine maintenance. While the server is suspended, it remains running and available for all administrative actions.

Prerequisites

Configure the Relay Server Outbound Enabler (RSOE) for Unwired Server in order to enable the suspend and resume server functions.

Task

1. In the Sybase Control Center navigation pane, click **Servers** to display the servers list.
2. Select one or more servers in this list.
3. Choose an appropriate process:
 - To suspend the server, click **Suspend**. Wait for about 1 minute, and click "refresh" button. When the server status changes from `suspend pending` to `suspended`, you can then perform the administration actions you required.
 - To then resume the server, click **Resume**.

As the server suspends and resumes, progress messages display in the Server Console pane.

Pinging a Server

Ping a server to test the availability of backend server connectivity and verify the server state (for example, started or stopped). By default ping uses whichever Internet Inter-ORB Protocol call you configured (IIOPS by default) to test if a server's connection is available.

1. In the left navigation pane, expand the **Servers** folder and select a server.
2. Select the **General** tab.
3. Click **Ping**.

The result displays in the console area.

Checking Unwired Server Status

Verify whether a server is running, stopped, or suspended.

1. In the left navigation pane, select **Servers**.
2. In the right administration pane, select the **General** tab.
3. In the Status column, check the server status corresponding to the server you are administering: running, stopped, suspended, suspend pending, or resume pending.
4. Use the controls in the administration console to start, stop, restart, suspend, or resume the server, as required.

Server Properties

Server properties let administrators manage server configuration settings to ensure smooth data exchange between the server and client. You can configure administration port, replication, and messaging properties in the Server Configuration node of Sybase Control Center.

Note: Properties you configure for an Unwired Server are cluster-affecting. Therefore, to make sure they are propagated correctly, Sybase recommends that you set them only on a primary cluster server.

General Server

Configure properties and security profiles for Unwired Server management and communication ports. These ports process incoming replication synchronization, administration, and data change notification requests. You must secure data transmission over management and DCN communication ports by creating and assigning an SSL configuration to the ports. You can also configure Unwired Server performance properties.

Unwired Server Management Ports

Management ports in Unwired Server process incoming administration connection requests from Sybase Control Center. Management ports use IIOPS by default, though IIOP can be configured as well.

If you choose an unencrypted administration channel, simply reconfigure the port and change the profile used. You must then ensure that the managed resource properties Sybase Control Center uses for outbound requests match that of Unwired Servers.

Note: If you change the profile from one that requires server authentication only (for example, the built-in profile named **default**) to one that requires mutual authentication (for example, as the built-in profile named `default_mutual`), you must have already saved the necessary SSL certificates to both the Unwired Server and the Sybase Control Center keystores and truststores. See *Security* for details.

Additionally, if you are using Sybase Control Center in a development/test environment, you must also configure the Unwired WorkSpace on all development computers accessing the development Unwired Server to also save the required certificates to the java keystore. Sybase does not recommend that you use mutual authentication for the management port.

Configuring Security Profiles

Configure security profiles to secure communication between Unwired Server administration and DCNs.

Prerequisites

Before creating a security profile, ensure that you possess digital certificates that have been verified and signed by third-party trusted authorities, as well as import required certificates in to the Unwired Server keystore.

Task

1. In the left navigation pane, expand the **Servers** folder and select a server.
2. Select **Server Configuration**.
3. In the right administration pane, select the **General** tab.
4. From the menu bar, select **SSL Configuration**.
5. Create a new Security Profile:
 - a) Name the security profile.
 - b) Enter the case sensitive certificate alias for the profile (defined in the server keystore).
 - c) Select the Authentication option.
6. Click **Save**.
7. In the server restart dialog, click **OK**.
8. Restart the server for these changes to take effect.

Next

Use the profile to encrypt administration and DCN ports.

Configuring SSL Properties

Configure SSL certificates and security profiles to facilitate Secure Sockets Layer (SSL) encryption for communication ports in Unwired Platform.

Prerequisites

Ensure you have set up the server environment before you configure a security profile as part of the server configuration. For more information, see *Encrypting Synchronization with SSL for Replication* in the *Security* guide.

Task

Note: If you enable SSL for one node in a cluster, you must enable it for all other nodes, since the secure connection is established at the cluster level. The port numbers for each node are not dependent; that is, they can be identical or unique.

Defining Certificates for SSL Encryption

For the primary server, specify keystore and truststore certificates to be used for SSL encryption of Unwired Platform communication ports. All security profiles use the same keystore and truststore.

For secondary Unwired Servers, SSL properties are synchronized from the primary server. Therefore for secondary servers, these properties are still visible, but cannot be edited.

1. In the left navigation pane, expand the **Servers** folder and select a server.
2. Select **Server Configuration**.
3. In the right administration pane, select the **General** tab.
4. From the menu bar, select **SSL Configuration**.
5. To configure SSL encryption for all security profiles, complete these fields:
 - **Keystore Location** – the full path name indicating the location where the keys and certificates are stored. Certificates used for administration and data change notification ports are stored in the keystore. The path should be relative to `<Unwired Platform_InstallDir>\UnwiredPlatform-XX\Servers\UnwiredServer`.
 - **Keystore Password** – the password that secures the key store.
 - **Truststore Location** – the full path name for the public key certificate storage file. The Certificate Authority (CA) certificates used to sign certificates store their public keys in the truststore. The path should be relative to `<Unwired Platform_InstallDir>\UnwiredPlatform-XX\Servers\UnwiredServer`.
 - **Truststore Password** – the password that secures the truststore.
6. Click **Save**.

Next

Create an SSL security profile that uses the selected certificates.

Creating an SSL Security Profile in Sybase Control Center

Security profiles define the security characteristics of a client/server session. Assign a security profile to a listener, which is configured as a port that accepts client connection requests of various protocols. Unwired Server uses multiple listeners. Clients that support the same characteristics can communicate to Unwired Server via the same port defined in the listener.

Note: A security profile can be used by one or more servers in a cluster, but cannot be used by multiple clusters.

1. In the left navigation pane, expand the **Servers** folder and select a server.
2. Select **Server Configuration**.

3. In the right administration pane, select the **General** tab.
4. From the menu bar, select **SSL Configuration**.
5. In the **Configure security profile table**:
 - a) Enter a name for the security profile.
 - b) Enter a certificate alias. This is the logical name for the certificate stored in the keystore.
 - c) Select an authentication level:

If the security profile authenticates only the server, then only the server must provide a certificate to be accepted or rejected by the client. If the security profile authenticates both the client and the server, then the client is also required to authenticate using a certificate; both the client and server will provide a digital certificate to be accepted or rejected by the other.

Profile	Authenticates	Cipher suites
intl	server	<ul style="list-style-type: none"> • SA_EX-PORT_WITH_RC4_40_MD5 • RSA_EX-PORT_WITH_DES40_CBC_SHA
intl_mutual	client/server	<ul style="list-style-type: none"> • RSA_EX-PORT_WITH_RC4_40_MD5 • RSA_EX-PORT_WITH_DES40_CBC_SHA
strong	server	<ul style="list-style-type: none"> • RSA_WITH_3DES_EDE_CBC_SHA • RSA_WITH_RC4_128_MD5 • RSA_WITH_RC4_128_SHA
strong_mutual	client/server For example, this is the required option for mutual authentication of Unwired Platform and Gateway.	<ul style="list-style-type: none"> • RSA_WITH_3DES_EDE_CBC_SHA • RSA_WITH_RC4_128_MD5 • RSA_WITH_RC4_128_SHA

Profile	Authenticates	Cipher suites
domestic	server	<ul style="list-style-type: none"> • RSA_WITH_3DES_EDE_CBC_SHA • RSA_WITH_RC4_128_MD5 • RSA_WITH_RC4_128_SHA • RSA_WITH_DES_CBC_SHA • RSA_EX-PORT_WITH_RC4_40_MD5 • RSA_EX-PORT_WITH_DES40_CBC_SHA • TLS_RSA_WITH_NULL_MD5 • TLS_RSA_WITH_NULL_SHA
domestic_mutual	client/server	<ul style="list-style-type: none"> • RSA_WITH_3DES_EDE_CBC_SHA • RSA_WITH_RC4_128_MD5 • RSA_WITH_RC4_128_SHA • RSA_WITH_DES_CBC_SHA • RSA_EX-PORT_WITH_RC4_40_MD5 • RSA_EX-PORT_WITH_DES40_CBC_SHA • RSA_WITH_NULL_MD5 • RSA_WITH_NULL_SHA

6. Click **Save**.

7. From the **Communication Ports** menu, assign the security profile to the desired management or communication ports.

Next

If you configure a secure port on one server, you must enable it on every node in the cluster, then restart all servers in the cluster to commit the configuration changes.

Enabling OCSP

(Optional) Enable OCSP (Online Certificate Status Protocol) to determine the status of a certificate used to authenticate a subject: current, expired, or unknown. OCSP configuration is enabled as part of server level SSL configuration. OCSP checking must be enabled if you are using the CertificateAuthenticationLoginModule and have set Enable revocation checking to true.

Enable OCSP for an Unwired Server when configuring SSL.

Administer

1. To enable OCSP when doing certificate revocation checking, check **Enable OCSP**.
2. Configure the responder properties (location and certificate information):

Responder Property	Details
URL	A URL to responder, including its port. For example, <code>https://ocsp.example.net:80</code> .
Certificate subject name	The subject name of the responder's certificate. By default, the certificate of the OCSP responder is that of the issuer of the certificate being validated. Its value is a string distinguished name (defined in RFC 2253), which identifies a certificate in the set of certificates supplied during cert path validation. If the subject name alone is not sufficient to uniquely identify the certificate, the subject value and serial number properties must be used instead. When the certificate subject name is set, the certificate issuer name and certificate serial number are ignored. For example, <code>CN=MyEnterprise, O=XYZCorp</code> .
Certificate issuer name	The issuer name of the responder certificate. For example, <code>CN=OCSP Responder, O=XYZCorp</code> .
Certificate serial number	The serial number of the responder certificate.

Replication

Replication synchronization involves synchronization between Unwired Server and a replication-based mobile device application. Synchronization keeps multiple variations of the data set used by a device application in coherence with one another by reconciling differences in each. Reconciling differences before writing updates back to the enterprise information server (EIS) maintains data integrity.

For replication synchronization, configure the corresponding port to receive incoming synchronization requests from devices, as well as set up configuration to enable push notification messages to the device when data changes in CDB. In a typical environment, client applications running on devices will connect to the synchronization port via Relay Server and Relay Server Outbound Enabler (RSOE). In those cases, the HTTP port will be used.

Configuring a Replication Listener

(Not applicable to Online Data Proxy) Configure the port to receive synchronization requests from client devices. If you are using push synchronization, then also configure synchronization listener properties.

Prerequisites

A secure synchronization stream uses SSL or TLS encryption. Both TLS and SSL require production-ready certificates to replace the default ones installed with Unwired Platform. Ensure that you possess digital certificates verified and signed by third-party trusted authorities. See *Encrypting Synchronization for Replication Payloads* in *Security*.

Task

1. In the left navigation pane, expand the **Servers** folder and select the primary server.

Note: You can only configure the encryption properties on the primary Unwired Server. Secondary servers will inherit the values, where they become read-only.

2. Select **Server Configuration**.
3. In the right administration pane, click the **Replication** tab.
4. If push synchronization is being added to the application, select **Listener** from the menu bar.
5. Select the protocol and port you require:
 - If you do not require SSL encryption, choose **Port**. Sybase recommends this option if you do not require a secure communication stream for synchronization. By default, the port for HTTP is 2480.
 - To encrypt the HTTP stream with SSL, choose **Secure port**. By default, the port for HTTPS is 2481. The "Secure Sync Port" properties in the "Optional Properties" table can be used to review and set the server identity and public certificate for the secure synchronization port. See below.
6. Configure these properties:
 - **Synchronization Cache Size** – sets the maximum cache size for the synchronization port. The default is 50MB.
 - **Thread Count** – sets the number of worker threads used for synchronization. The default is 10. If you experience performance issues, you may need to increase this value.
7. (Optional) Expand the optional properties section to configure additional properties for E2EE with TLS, HTTPS with SSL, and synchronization server startup options:

Note: Leave E2E Encryption values blank to disable end-to-end encryption.

- E2E Encryption Certificate Password – set the password to unlock the encryption certificate.
- E2E Encryption Certificate – specify the file containing the private key that acts as the identity file for Unwired Server.
- E2E Encryption Type – specify the asymmetric cipher used for key exchange for end-to-end encryption. You can only use RSA encryption.
- Secure Sync Port Certificate – identifies the location of the security certificate used to encrypt and decrypt data transferred using SSL.
- Secure Sync Port Certificate Password – is used to decrypt the private certificate listed in certificate file. You specify this password when you create the server certificate for SSL.
- Secure Sync Port Public Certificate – specify the file containing the SSL public key that acts as the identity file for synchronization port.
- Trusted Relay Server Certificate – if Relay Server trusted certificate is configured for HTTPS connections encrypted with SSL, identifies the public security certificate location.
- User Options – sets the command line options for starting the synchronization server. These options are appended the next time the synchronization server starts. These are the available user options:

Option	Description
@ [<i>variable</i> <i>filePath</i>]	Applies listener options from the specified environment variable or text file.
-a <value>	Specifies a single library option for a listening library.
-d <filePath>	Specifies a listening library.
-e <deviceName>	Specifies the device name.
-f <string>	Specifies extra information about the device.
-gi <seconds>	Specifies the IP tracker polling interval.
-i <seconds>	Specifies the polling interval for SMTP connections.
-l <"keyword=value;...">	Defines and creates a message handler.
-m	Turns on message logging.
-ni	Disables IP tracking.
-ns	Disables SMS listening.
-nu	Disables UDP listening.

Option	Description
-o <filePath>	Logs output to a file. Note: Ensure that you enter the absolute file path for this property.
-os <bytes>	Specifies the maximum size of the log file.
-p	Allows the device to shut down automatically when idle.
-pc [+ -]	Enables or disables persistent connections.
-r <filePath>	Identifies a remote database involved in the responding action of a message filter.
-sv <scriptVersion>	Specifies a script version used for authentication.
-t [+ -] <name>	Registers or unregisters the remote ID for a remote database.
-u <userName>	Specifies a synchronization server user name.
-v [0 1 2 3]	Specifies the verbosity level for the messaging log.
-y <newPassword>	Specifies a new synchronization server password.

Do not use the User Options property in Sybase Control Center to pass in these options: -c, -lsc, -q, -w, -x, -zs.

For more information on synchronization server command line options, see *MobiLink Listener options for Windows devices* (<http://infocenter.sybase.com/help/topic/com.sybase.help.sqlanywhere.12.0.1/mlsync/ms-listener-s-3217696.html>) in the *SQL Anywhere® 12.0.1* online help.

8. Click **Save**.

Messaging

Messaging is a synchronization method used to maintain data integrity on device client applications. It uses a JMS service to upload and download data changes to and from the Unwired Server cache database. Messaging-based synchronization ports implement a strongly encrypted HTTP-based protocol using a proprietary method.

Configure messaging in the Messaging tab of the Server Configuration node for the particular server you are administering.

Configuring Messaging Synchronization Properties

(Not applicable to Online Data Proxy) Configure one or more synchronization ports to receive service requests from devices.

1. In the left navigation pane, expand the **Servers** folder and select a server.
2. Select **Server Configuration**.
3. In the right administration pane, click the **Messaging** tab, and select **Listener**.
4. Enter the synchronization port number. The default is 5001.
5. (Optional) Select **Listen on multiple ports** and enter the additional port numbers.

Depending on your environment, listening on multiple synchronization ports may provide greater flexibility and reliability. High activity on particular ports, such as virus detection and data inspection, may result in dropped packets or connections if alternate ports are unavailable. When multiple ports are configured, all messaging traffic is still funneled to a single listener.

6. Click **Save**.

Configuring Apple Push Settings

Create a new Apple Push Notification Service (APNS) configuration that specifies the application, security certificate, and ports that the service uses.

Apple push notifications alert offline iPhone users to the availability of new items awaiting retrieval on Unwired Server. Push uses an IP connection only long enough for the Send/Receive data exchange to complete. The feature overcomes network issues with always-on connectivity and battery life consumption on 3G networks.

For more information on end-to-end iPhone application development and provisioning, search for *Apple Provisioning for iOS* in *System Administration*.

Note: When configuring the Apple Push Notification Service, change the push gateway, push gateway port, feedback gateway, and feedback gateway port values only when configuring notifications in a development environment. To enable Apple push notifications, the firewall must allow outbound connections to Apple push notification servers on default ports 2195 and 2196.

1. In the left navigation pane, expand the **Servers** folder and select a server.
2. Select **Server Configuration**.
3. In the Messaging tab, select **Apple Push Configuration**.
4. Click **New**.
5. Enter the **Application name**. This name corresponds to the Product Name specified in Xcode.
6. Select one of:

- **Use existing certificate** – use a security certificate file that already exists on the server. When you select this option, the list of available certificates appears in the **Certificate name** menu.
 - **Use new certificate** – create a new security certificate. When you select this option, you must provide information to create a named security certificate file on the server.
7. If you selected **Use existing certificate**:
 - a) Select the desired certificate from the list.
 - b) Enter and confirm the certificate password.
 8. If you selected **Use new certificate**:
 - a) Enter a name for the new certificate.
 - b) Specify a Base64-encoded string by choosing one of these:
 - **Browse from file** – select a security certificate file on the server that contains the Base64-encoded string.
 - **Base64-encoded string** – manually enter the Base64-encoded string.
 - c) If you selected a file from the server for the Base64-encoded string, you can overwrite the existing certificate file with the details you specify during new certificate creation. To do so, select the box adjacent to **Overwrite existing certificate**.
 - d) Enter and confirm the certificate password.
 9. Click **OK**.

Configuring BlackBerry Push Settings

Create a new BlackBerry Push Service configuration that specifies the delivery method for the service users.

BlackBerry push notifications alert offline users to the availability of new items awaiting retrieval on Unwired Server. Push uses an IP connection only long enough for the Send/Receive data exchange to complete. BlackBerry Push notifications overcome issues with always-on connectivity and battery life consumption over wireless networks.

1. In the left navigation pane, expand the **Servers** folder and select a server.
2. Select **Server Configuration**.
3. In the Messaging tab, select **BlackBerry Push Configuration**.
4. (Optional) Configure the BES response portioning value to fine tune the amount of HTTP traffic the BES MDS server accepts from Unwired Server.
 - a) Select the unit: MB, KB or bytes.
 - b) Type an integer value.

By default, this setting is configured to 100 KB in Unwired Server, to accommodate older BES servers. Newer BES servers are configured with a 1MB default. If you have a newer BES server, improve performance by setting a higher value. Experiment with different values to fine tune the correct value for your system configuration.

5. Select an existing BlackBerry push configuration, and click **Properties** to view its current settings.
6. (Optional) create a new BlackBerry push configuration.
 - a) Click **New**.
 - b) Enter the **Name** of the configuration.
 - c) Enter the **URL** of the delivery service host. The URL is in the format: `http://<DNS or IP address>:<port number>`. The default port is 8080.

Note: In a development environment, if the BlackBerry simulator and MDS simulator are on different machines, enter the URL of the MDS simulator.

- d) (Optional) Enter a **User name** and **Password** required to access the URL. Confirm the password.
- e) BES (BlackBerry Enterprise Server) is selected by default.
- f) (Optional) Click PAP to enable the Push Access Protocol (PAP), if you are using a Push Proxy Gateway to deliver messages.
- g) Click **OK**.

Configuring Unwired Server Performance Properties

To optimize Unwired Platform performance, configure the thread stack size, maximum and minimum heap sizes, user options, and inbound and outbound messaging queue counts.

1. In the left navigation pane, expand the **Servers** folder and select a server.
2. Select **Server Configuration**.
3. In the right administration pane, select the **General** tab.
4. From the menu bar, select **Performance Configuration**.
5. Configure these properties, as required:
 - Host Name – the name of the machine where Unwired Server is running (read only).
 - Thread Stack Size – the JVM `-Xss` option.
 - Minimum Heap Size – the minimum size of the JVM memory allocation pool, in megabytes. For production recommendations on this value, see *Unwired Server Replication Tuning Reference* in *System Administration*.
 - Maximum Heap Size – the maximum size of the JVM memory allocation pool, in megabytes. For production recommendations on this value, see *Unwired Server Replication Tuning Reference* in *System Administration*.
6. (Optional) Expand the **Show optional properties** section and configure these properties, as required:
 - Maximum Number of In Memory Messages – specify the number of in-memory messages to allow.
 - User Options – other JVM options. For example, you can enable JVM garbage collection logging by setting `-XX:+PrintGCDetails`. Or you can set the

permanent space which allocated outside of the Java heap with DJC_JVM_MAXPERM; the maximum perm size must be followed by K, M, or G, for example, – XX:MaxPermSize=512M. Note that DJC_JVM_MAXPERM is not visible to Sybase Control Center.

- Inbound Messaging Queue Count – the number of message queues used for incoming messages from the messaging-based synchronization application to the server. Sybase recommends a choose a value that represents at least 10% of active devices.
- Outbound Messaging Queue Count – the number of message queues used for outbound messages from the server to the messaging-based synchronization application. Sybase recommends a choose a value that represents at least 50% of active devices. However, if you are running 32-bit operating system, do not exceed a value of 100% of active devices.
- Subscribe Bulk Load Thread Pool Size – the maximum number of threads allocated to initial bulk load subscription operations. The default value is five. Setting the thread pool size too high can impact performance.

Note: If you increase either queue count property, ensure you also increase the MaxThread property in the <hostname>_iiopl.properties file.

7. Click **Save**.

Saving and Refreshing an Unwired Server Configuration

Refreshing an Unwired Server configuration displays the latest effective configuration information.

After successfully saving a server configuration, refresh the configuration to display the most recent updates. To commit these changes to the server, restart the server before saving subsequent updates. The refresh function must be used in conjunction with a server restart for the displayed configuration to be applied.

If you refresh the configuration in between two sets of saved configuration changes without injecting a server restart following the refresh, only the second set of changes are committed and consequently displayed as the current set of properties used by Unwired Server.

Note: Follow the steps in exactly the order they appear. Otherwise, configuration changes will be lost.

1. Reconfigure Unwired Server as required.
2. Click **Save**.
3. Click **Refresh** to display original values; the recent'y saved changes are not displayed.
4. Restart Unwired Server to commit those changes, using the method you prefer for server restarts.
5. In the left navigation pane, expand the **Servers** folder and select a server.
6. Select **Server Configuration**.

7. In the right administration pane, select the appropriate tab and click **Refresh**.
Current server configuration properties committed with the restart action appear.
8. Make the next set of configuration changes, as required.

Reviewing Pending Changes

As you configure Unwired Server with Sybase Control Center, changes that require a server restart are aggregated to the **Pending Changes** tab for the server name you are currently administering.

Changes listed in this window require a server restart before they take effect.

1. In the left pane, click the Unwired server you are currently logged into.
2. Click **Pending Changes**.
3. Review all listed changes that are pending.
4. If the changes are valid, click **Restart** to commit the changes.
5. A confirmation message to continue appears. Confirm that you want to restart the server.
6. Review Unwired Server status messages on the **General** tab to ensure that the server has restarted and changes have been committed successfully. If the update is successful, the bolded text and asterisk (*) are also removed from the respective server name in the left navigation pane.

Applying Multiple Unwired Server Configuration Changes

A server restart writes the changes made in Sybase Control Center to the appropriate Unwired Server configuration file. To apply multiple server configuration changes with a single server restart, you cannot make consecutive conflicting updates or refresh the configuration in between saved changes.

Consider these important points when applying multiple changes to an Unwired Server configuration:

- Failure to save a configuration change prior to restarting Unwired Server results in configuration changes being lost.
- Failure to restart Unwired Server after saving a configuration change results in changes being uncommitted; Unwired Server instead uses the values that currently exist in the configuration file (that is, previous configuration properties and values).
- Cumulative saved changes are applied successfully upon server restart as long as these updates do not conflict. Attempting to save two conflicting sets of changes fails. In this case, inject a server restart in between each saved change to ensure that the required updates are propagated across the server.
- Refreshing the server configuration displays the latest successfully saved configuration information. If you click Refresh in between two sets of saved changes, only the most recent saved updates are applied during a server restart.

When you must make multiple changes to the same component of the Unwired Server configuration, follow this procedure:

Note: Follow the steps in exactly the order they appear. Do not use the Refresh function in between saved changes. Otherwise, configuration changes will be lost.

1. Make the first set of configuration changes, as required.
2. Click **Save**.
A confirmation message appears in the administration console indicating the success or failure of the save.
3. Make the second set of configuration changes, as required.
4. Click **Save**.
A confirmation message appears in the administration console indicating the success or failure of the save. If the save is unsuccessful, restart the server before reattempting these updates.
5. Restart Unwired Server to commit the changes in steps 1 and 3, using the method you prefer for server restarts.
6. In the left navigation pane, expand the **Servers** folder and select a server.
7. Select **Server Configuration**.
8. In the right administration pane, select the appropriate tab and click **Refresh**.
Current server configuration properties committed with the restart action appear.

Viewing Unwired Server Properties

View information, including host names, port numbers, version, and file location, to help you manage an Unwired Server and its components.

1. In the left navigation pane, expand the **Servers** folder and select a server.
2. In the right administration pane, select the **Properties** tab.
3. Review Unwired Server properties.

Relay Server Outbound Enabler

The Outbound Enabler (RSOE) runs as an Unwired Server process and manages communication between the Unwired Server and a Relay Server.

Each RSOE maintains connections to each Relay Server in a Relay Server farm. The RSOE passes client requests to the Unwired Server on its Replication or Messaging port. Unwired Server sends its response to the RSOE, which forwards it to the Relay Server, to be passed to the client.

As an Unwired Server process, the RSOE always starts when Unwired Server starts. Unwired Server monitors the process to ensure it is available. If an RSOE fails for any reason, Unwired Server restarts it automatically.

Note: Sybase recommends three RSOE processes each, for both Replication and Messaging ports.

Loading and Unloading HTTPS Certificates for RSOE

Load HTTPS certificates for the RSOE to add it to the Unwired Server node .

Note: You must use only RSA certificates.

If the Web server (Relay Server host) already uses a certificate signed by a CA for HTTPS connections, you do not need to perform this task.

1. In the navigation pane, click **Servers > ServerNode > Server Configuration**.
2. In the administration pane, select the Outbound Enabler tab, then click **Certificate Files**.
3. Choose the action you want to perform:
 - To add a new certificate, click +. Browse and select the .CRT file to upload, then click **Open**.
 - To replace a certificate in the store, select **Overwrite the certificate file**, then click +.
 - To delete a certificate from the store, select the filename and click **X**.
4. When certificate management tasks are complete, click **OK**.

Setting Up RSOE

Set up one or more RSOEs for each Unwired Server identified in a Relay Server configuration. The configured values are saved in the cluster database.

Configuring RSOE General Properties

Set general RSOE configuration properties to define the context in which the RSOE process operates.

1. In the navigation pane, click **Servers > <ServerNode> > Server Configuration**.
2. In the administration pane, select the Outbound Enabler tab, then click **New**.
3. Specify these property values:
 - **Farm type** – select the type of request managed by the Relay Server, Replication or Messaging protocol.
 - **Unwired sever port** – select the port on which RSOE will manage requests.
 - **Relay server host** – select either the host name of the Relay Server, or the host name of the load balancer (if any).
 - **Relay server port** – select the Relay Server HTTPS port.
 - **Unwired server farm** – select the string that identifies the Unwired Server cluster, for which the Relay Server manages requests. This property is case-sensitive, and it must match the value in the Relay Server configuration.

- **Server node ID** – select the string that identifies the Unwired Server in the cluster. This property is case-sensitive, and it must match the value in the Relay Server configuration.

4. Click **Next**.

Configuring RSOE Connection Settings

Set connection configuration properties for an RSOE. These properties define the RSOE connection to the Relay Server.

1. Specify these property values:

- **Http user name** – select the user name for RSOE authentication on the Web server (Relay Server host).
- **Http password** – enter the password for RSOE authentication on the Web server.

2. If RSOE connections to the Relay Server must pass through an Internet proxy server, specify these property values:

- **Proxy server** – select the Internet proxy server.
- **Http proxy user** – select the user name for RSOE authentication on the proxy server.
- **Http proxy password** – type the password for RSOE authentication on the proxy server.

3. Specify these property values:

- **Certificate file** – select this option and choose the .CRT file used to authenticate the RSOE to relay server. You can only choose this file if you have already loaded it into the Unwired Server certificate store.
- **Trusted certificate** – if the certificate file includes multiple certificates, choose whether to trust a single certificate or all of them.

Configuring RSOE Start Options

Configure start options for RSOE.

1. Enable an option:

- a) Check the box that corresponds to each name.
- b) Set a value. If you check the box but set no value for the option, the default is used.

2. Click **OK**.

3. Ensure the process starts by checking the Status column of the Outbound Enablers tab.

Outbound Enabler Start Options Reference

Review available Outbound Enabler start options. These options affect Outbound Enabler logging. Each Outbound Enabler has its own log file that you can retrieve in Sybase Control Center.

Option	Default	Description
Verbosity level	0	Sets log file verbosity values: <ul style="list-style-type: none"> • 0 – Log errors only. Use this logging level for deployment. • 1 – Session level logging. This is a higher level view of a session. • 2 – Request level logging. Provides a more detailed view of HTTP requests within a session. • 3 - 5 – Detailed logging, used primarily for technical support.
Reconnect delay	5	Delay before retry after connection fails.
Maximum output file size	10KB	Maximum log file size.
Truncate log file	None	Option to delete the log file at RSOE startup.

Generating the Relay Server Outbound Enabler Configuration File

To quickly and easily replicate a common Outbound Enabler (RSOE) configuration to multiple hosts, generate an RSOE configuration file.

Administrators can use Sybase Control Center to configure an initial RSOE for development. Once a configuration proves valid and stable, the administrator can generate the RSOE configuration file, then use `regRelayServer.bat` to apply it to Unwired Server hosts.

1. In the navigation pane, click the Unwired Server cluster name.
2. In the administration pane, click the **Relay Servers** tab.
3. Click **Generate**.
4. Choose **Relay server outbound enabler configuration XML file**, then click **Next**.
5. Select an output target for the file.

6. Click **Finish**.

Managing Configured RSOEs

Manage RSOE instances you have configured.

Retrieving RSOE Logs

You can retrieve one RSOE log at a time, from the Unwired Server host, and copy it to another location. You cannot retrieve an empty RSOE log file.

1. In the navigation pane, click **Servers > <ServerNode> > Server Configuration**.
2. In the administration pane, select the Outbound Enabler tab, and select the RSOE instance.
3. Click **Retrieve Log**, then **Next**, then **Finish** to save the log and choose the target location for the file.

Viewing or Editing RSOE Properties

View or edit configuration properties for a selected RSOE instance.

Relaunching the RSOE Configuration Wizard

Relaunch the Outbound Enabler Configuration wizard to create a new RSOE configuration.

1. In the navigation pane, click **Servers > <ServerNode> > Server Configuration**.
2. In the administration pane, select the Outbound Enabler tab.
3. Select the RSOE instance, then click **Properties**.

Configuring RSOE General Properties

Set general RSOE configuration properties to define the context in which the RSOE process operates.

1. In the navigation pane, click **Servers > <ServerNode> > Server Configuration**.
2. In the administration pane, select the Outbound Enabler tab, then click **New**.
3. Specify these property values:
 - **Farm type** – select the type of request managed by the Relay Server, Replication or Messaging protocol.
 - **Unwired sever port** – select the port on which RSOE will manage requests.
 - **Relay server host** – select either the host name of the Relay Server, or the host name of the load balancer (if any).
 - **Relay server port** – select the Relay Server HTTPS port.
 - **Unwired server farm** – select the string that identifies the Unwired Server cluster, for which the Relay Server manages requests. This property is case-sensitive, and it must match the value in the Relay Server configuration.

- **Server node ID** – select the string that identifies the Unwired Server in the cluster. This property is case-sensitive, and it must match the value in the Relay Server configuration.

4. Click **Next**.

Configuring RSOE Connection Settings

Set connection configuration properties for an RSOE. These properties define the RSOE connection to the Relay Server.

1. Specify these property values:

- **Http user name** – select the user name for RSOE authentication on the Web server (Relay Server host).
- **Http password** – enter the password for RSOE authentication on the Web server.

2. If RSOE connections to the Relay Server must pass through an Internet proxy server, specify these property values:

- **Proxy server** – select the Internet proxy server.
- **Http proxy user** – select the user name for RSOE authentication on the proxy server.
- **Http proxy password** – type the password for RSOE authentication on the proxy server.

3. Specify these property values:

- **Certificate file** – select this option and choose the .CRT file used to authenticate the RSOE to relay server. You can only choose this file if you have already loaded it into the Unwired Server certificate store.
- **Trusted certificate** – if the certificate file includes multiple certificates, choose whether to trust a single certificate or all of them.

Configuring RSOE Start Options

Configure start options for RSOE.

1. Enable an option:

- a) Check the box that corresponds to each name.
- b) Set a value. If you check the box but set no value for the option, the default is used.

2. Click **OK**.

3. Ensure the process starts by checking the Status column of the Outbound Enablers tab.

Deleting RSOE Configurations

Delete an RSOE configuration to remove the configuration properties from the cluster database.

1. In the navigation pane, click **Servers > <ServerNode> > Server Configuration**.

2. In the administration pane, select the Outbound Enabler tab, and select the RSOE instances.
3. Stop the RSOE instances and click **Delete**.
4. Click **OK**.

Refreshing the RSOE List

Refresh the RSOE list to display current information about deployed and configured RSOE instances.

1. In the navigation pane, click **Servers > <ServerNode> > Server Configuration**.
2. In the administration pane, select the Outbound Enabler tab, and click **Refresh**.

Starting and Stopping RSOE

Start and stop an RSOE process as needed. All configured RSOEs are started by default when the Unwired Server starts.

1. In the navigation pane, click **Servers > ServerNode > Server Configuration**.
2. In the administration pane, select the Outbound Enabler tab, select the RSOE instances, and click **Start** or **Stop**.

Configuring Proxy Server Settings for an Outbound Enabler

(Applies only to Online Data Proxy) Configure an Outbound Enabler to work with an Internet proxy server, when connections to the Relay Server must pass through a proxy server.

1. In the navigation pane, click **Servers><ServerNode> > Server Configuration**.
2. In the administration pane, select the **Outbound Enabler** tab.
3. Click **Proxy** .
4. Define a list of required proxy servers:
 - a) To add a new server connection, type **Host** and **Port** values, then click **+**.
 - b) To remove an existing connection, select the server, then click **X**.
 - c) To edit an existing connection, click an appropriate cell and re-enter or modify the current value.
5. Define a proxy user for a selected server:
 - a) Select a server from the list.
 - b) To add a new user, enter a User name and password then click **+**.
 - c) To remove an existing user, select the name, then click **X**.
 - d) To edit an existing user, click an appropriate cell and re-enter or modify the current value.
6. Click **OK**.

Outbound Enabler Tab Reference

Understand the columns of data displayed in the Outbound Enabler tab for an Unwired Server node.

Column Name	Displays
Server Node ID	the string that identifies the Unwired Server in the cluster. This property is case-sensitive, and it must match the value in the Relay Server configuration.
Unwired Server Port	the port on which Outbound Enabler manages requests.
Farm Type	the type of request managed by the Relay Server, Replication or Messaging protocol.
Unwired Server Farm	the string that identifies the Unwired Server cluster, for which the Relay Server manages requests. This property is case-sensitive, and it must match the value in the Relay Server configuration.
Relay Server Host	either the host name of the Relay Server, or the host name of the load balancer (if any).
Status	current state of the Outbound Enabler process: stopped, running, or error.
Status Description	<p>additional details on the state of the Outbound Enabler. If you receive one of these messages, follow the documented recommendation:</p> <ul style="list-style-type: none"> • Unknown error state – Check the log for additional details. • Failed to connect Unwired Server, retrying... – Check the Unwired Server port managed by the Outbound Enabler. • Unauthorized. – Check the security token of Outbound Enabler. • Unrecognized farm or server node ID. – The string that identifies the Unwired Server cluster or server node in the Outbound Enabler configuration does not match the Relay Server configuration. • Please check the relay server host and port or Failed to create I/O stream to the relay server – If you use HTTPS port, check to see if the certificate file is valid. • Relay server service unavailable. – Check if the Relay Server is properly configured, or if any internal errors are logged. • Relay server not found. – Either the Relay Server is not yet deployed, or the URL suffix is wrong. • Bad request. – Check the syntax of the URL suffix. • Error writing HTTP headers – Check if the trusted certificate is valid, and verify the URL suffix syntax. Something may be misformatted.
Certificate File	the certificate file uploaded to the Unwired Server certificate store.

Column Name	Displays
Log File	the name and location of the Outbound Enabler log file. The syntax of this filename is <code><nodeName>.RSOE<n>.log</code> . <code><n></code> is the primary key of the Outbound Enabler database record in the cluster database used by Unwired Platform.

Server Log

Server logs enable you to monitor system health at a high level, or focus in on specific issues by setting up filtering criteria using Sybase Control Center

These server logs are available:

- Unwired Server logs – collect data on Unwired Server health and performance by component, and retrieve data for all or specific searches. You can save and archive system logs, and manage log file size and rollover behavior.
- Messaging Server logs – create trace configurations for messaging modules, and retrieve trace data for all or specific messages. Export data for archive or further analysis.

Note: Properties you configure for an Unwired Server are cluster-affecting. Therefore, to make sure they are propagated correctly, Sybase recommends that you set them only on a primary cluster server.

Unwired Server Runtime Logging

Unwired Server logs collect runtime information from various embedded runtime components.

By default, all the components of the Unwired Server log are set at the INFO level, except the Other components, which are set at the WARN level. However, you can change this level as required. You should only use Sybase Control Center to set these logging values to ensure they are configured correctly. These values will be correctly transcribed to an internal file (that is, `<UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\UnwiredServer\Repository\logging-configuration.xml`).

You can view these Unwired Server logs in two ways:

- From Sybase Control Center – click **Servers > primaryServer > Log** in the left pane, and **Unwired Server > General** in the right pane.
The first 150 entries initially appear in the console area, so you may need to incrementally retrieve more of the log as required, by scrolling down through the log events.
- From a text editor – browse to and open one or more of the `<UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\UnwiredServer\logs\<hostname>-server.log` files. These files may be indexed, depending on how you configure the life cycle for the servers's log file.

Configuring Unwired Server Log Settings

Configure Unwired Server log properties to specify the amount of detail that is written to the log, as well as the duration of the server log life cycle. Server log properties are only set on the primary node; property settings on secondary nodes are read-only.

How changes are applied in a cluster depends on whether you are configuring a primary or secondary server. Sybase recommends you only configure log settings on the primary server. If you change the setting on a secondary server, the configuration is updated only for that server and is temporary (eventually the primary settings are propagated to all servers in the cluster).

Additionally, you should always use Sybase Control Center to configure server logs. If you manually edit the configuration file, especially on secondary servers in a cluster, the servers may not restart correctly once shut down.

1. In the Sybase Control Center left navigation pane, click **Servers > *primaryServer* > Log**, and in the right pane click **Unwired Server > Settings**.
2. The option "Start a new server log on server restart" is set by default. When selected, this option means a new version of the log file is created after server restart, and the old one is archived.
3. Set the server log size and backup behavior that jointly determine the server log life cycle.
 - a) Set the **Maximum file size**, in kilobytes, megabytes, or gigabytes, to specify the maximum size that a file can reach before a new one is created. The default is 10MB. Alternatively, select **No limit** to log all events in the same file, with no maximum size.
 - b) Set the **Maximum backup index** to determine how many log files are backed up before the oldest file is deleted. The index number you choose must be a positive integer between 1 and 65535. The default is 10 files. Alternatively, select **No limit** to retain all log files.
4. For each component, choose a log level:

Component	Default Log Level
MMS	Info
PROXY	Info
MSG	Info
Security	Info
Mobilink	Info
DataServices	Info
Other	Warn
DOEC	Info

Note: DOEC only appears if you run the DOEC installer after installing SUP cluster.

Log level	Messages logged
All	Complete system information
Trace	Finer-grained informational events than debug
Debug	Very fine-grained system information, warnings, and all errors
Info	General system information, warnings, and all errors
Warn	Warnings and all errors
Error	Errors only
Console	Messages that appear in the administration console only (when Unwired Server is running in non-service mode)
Off	Do not record any messages

5. Click **Save**.

Log messages are recorded as specified by the settings you choose. The log file is located in: `<UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\UnwiredServer\logs\<hostname>-server.log`.

Log life cycle default example

If you keep the default maximum file size and default index, an Unwired Server writes to the log file until 10MB of data has been recorded. As soon as the file exceeds this value, a new version of the log file is created (for example, the first one is `<hostname>-server.log.1`). The contents of the original log are backed up into this new file. When the `<hostname>-server.log` file again reaches its limit:

1. The contents of `<hostname>-server.log.1` are copied to `<hostname>-server.log.2`.
2. The contents of `<hostname>-server.log` are copied to `<hostname>-server.log.1`.
3. A new copy of `<hostname>-server.log` is created.

This rollover pattern continues until the backup index value is reached, with the oldest log being deleted. If the backup index is 10, then `<hostname>-server.log.10` is the file removed, and all other logs roll up to create room for the new file.

Viewing the Unwired Server Log

In text or grid view, use the vertical scroll bar to retrieve additional segments of the log file in 150 line increments. In grid view, up to 10 pages of the server log data is loaded in one request.

You can navigate to any page by using the **First**, **Prev**, **Next**, **Last**, and **Go to** controls. Use **View Details** open the actual log file and find the corresponding line.

There are also two search options you can use:

- **Basic search** – allows you to search by keyword, log level, first/last X number of lines in the log file.
- **Advanced search** – allows you to search by specific subcomponents, log level, exception, time range, and so on.

You can include backup logs in your search or retrieval. The option is not selected by default.

Searching Unwired Server Log Data

Filter server log data according to the criteria you specify.

1. In the Sybase Control Center left navigation pane, click **Servers > primaryServer > Log**, and in the right pane click **Unwired Server > General**.
2. Select **Show filter criteria** to display the search pane.
3. Select **Include backup logs** to display backup logs.
4. Select **Text view** or **Grid view** to specify how to display the logs.
5. Select **Basic search** to filter your search according to the specific string you enter in the search field. (Optional) You may also specify:
 - **Show** – specify first lines, last lines, or a keyword. If you are searching by first or last lines, you can enter any value up to a maximum of 1000 lines in the log. However, Sybase recommends that you provide a more manageable value to avoid severe performance degradation associated with this upper limit.
 - **Log level** – search only messages logged by the particular log level you select.
6. Select **Advanced search** to enter more specific search criteria, including:
 - **Component** – identify which component the log data belongs to: MMS, Proxy, MSG, Security, MobiLink™, DataServices, Other or DOEC.

Note: Set the log level for each component in the **Setting** tab. See *Configuring Server Log Settings* in Sybase Control Center online help.

- **Log level** – search only messages logged by the particular log level you select.
 - **Thread ID** – specify the ID name of the thread that logs the message you are searching.
 - **Logger name** – indicate the class name and instance of the logged component.
 - **Keyword** – indicate a value, file name, or other keyword by which to filter your search.
 - **Time period** – specify a start date, start time, end date, and end time.
7. Click **Retrieve**.
 8. To begin a new query, click **Reset** in the search panel and enter new search criteria.

Retrieving the Unwired Server Log

Update the information in the log console window.

1. In the Sybase Control Center left navigation pane, click **Servers > primaryServer > Log**, and in the right pane click **Unwired Server > General**.
2. To display the latest log data collected in the log file if time has elapsed since you last opened the log, click **Retrieve**.
3. (Optional) Select a row to view a single record in the detail pane. Additional columns may be available.

Deleting the Unwired Server Log

Clear old or unrequired server log data from the log file.

1. In the Sybase Control Center left navigation pane, click **Servers > primaryServer > Log**, and in the right pane click **Unwired Server > General**.
2. To delete all data from the log file and all backup log files, click **Delete**, then **OK**.

Messaging Server Runtime Logging

Messaging Server logs collect data that enables you to trace message handling from the cluster database to the device user, based on various trace settings.

You can configure trace settings for the primary server cluster in Sybase Control Center for each module. The settings are available to cluster servers through the shared data folder.

Configuring Messaging Server Log Settings

Configure trace configuration properties for modules to specify the amount of detail that is written to the log. Messaging Server log settings are cluster-wide, so changes made on the primary node are effective on all nodes.

How changes are applied in a cluster depends on whether you are configuring a primary or secondary server. Sybase recommends you only configure log settings on the primary server. If you change the setting on a secondary server, the configuration is updated only for that server and is temporary (eventually the primary settings are propagated to all servers in the cluster).

Note: The default settings may only need to change in case of technical support situations where, for diagnostic reasons, a request is made to configure the specific module(s) settings, and provide the request log. In all other cases, the administrator or developer should not need to change the settings.

Additionally, you should always use Sybase Control Center to configure server logs. If you manually edit the configuration file, especially on secondary servers in a cluster, the servers may not restart correctly once shut down.

Administer

1. In the Sybase Control Center left navigation pane, click **Servers > primaryServer > Log**, and in the right pane click **Messaging Server > Settings**.
2. Select Default, or one or more of the messaging service modules. Click **Show All** to show all modules.

Module	Description
Default	Represents the default configuration. The default values are used if optional fields are left blank in a module trace configuration. Required.
Device Management	Miscellaneous functions related to device registration, event notification, and device administration. Enable tracing for problems in these areas.
JMSBridge	The administrative interface for the JmsBridge configuration.
MO	This module handles the delivery of messages between the client and server, including synchronous function calls from client to server. Enable tracing for MO errors and message delivery issues.
MOCA	The Mobile Objects manager for server-to-device requests.
SUPBridge	SUPBridge is the communication module from the messaging server and the Unwired Server. Enable tracing for messaging communications errors.
TM	This module handles the wire protocol, including encryption, compression, and authentication, between the messaging server and clients. All communication between the client and the messaging server passes through TM. Enable tracing for authentication issues, TM errors, and general connectivity issues.
WorkflowClient	The WorkflowClient module.

3. Click **Properties**.
 - a) Enter trace configuration properties. If you selected multiple modules, a string of asterisks is used to indicate settings differ for the selected modules. You can select the option to view or change the property value for any module.

Property	Description
Module	Display only. Default, module name, or list of module names selected.
Description	(Optional) Custom description of the server module.
Level	Trace level for the module - DISABLED, ERROR, WARN, INFO, DEBUG, DEFAULT. If the default trace level is specified for the module, the module uses the trace level defined for Default. Required.
Max trace file size	(Optional) Maximum trace file size in MB. If the trace file size grows larger than the specified value, the trace file data is backed up automatically.
User name	(Optional) Only data for the specified user name is traced.
Application Connection ID	(Optional) Only data for the specified Application ID is traced.

b) Click **OK**.

Log files for each module are stored in folders of the same name located in:
`<UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers
 \UnwiredServer\logs.`

Viewing the Messaging Server Log

You can view results for one or more modules, or the Default. You can navigate to any page by using the **First**, **Prev**, **Next**, **Last**, and **Go to** controls.

Searching Messaging Server Log Data

Filter server log data according to the criteria you specify.

1. In the Sybase Control Center left navigation pane, click **Servers > primaryServer > Log**, and in the right pane click **Messaging Server > General**.
2. Click **Show filter**, and then select the search criteria:
 - **Max level** – search only messages logged by the particular log level you select. All messages up to that level are retrieved.
 - **Thread ID** – specify the ID name of the thread that logs the message you are searching.
 - **Contains** – enter a search string.
 - **Users** – select one or more users.
 - **Application connections** – select one or more application connections.

- **Modules** – select one or more modules.
 - **Time period** – specify a start date, start time, end date, and end time.
3. Click **Retrieve**.
 4. To begin a new query, click **Reset**.

Retrieving the Messaging Server Log

Update the information in the log console window.

1. In the Sybase Control Center left navigation pane, click **Servers > primaryServer > Log**, and in the right pane click **Messaging Server > General**.
2. To display the latest log data collected in the log file if time has elapsed since you last opened the log, click **Retrieve**.
3. (Optional) Select a row to view a single record in the detail pane. Additional columns may be available.

Exporting Messaging Server Log

Export retrieved trace information for archive or further analysis.

1. In the Sybase Control Center left navigation pane, click **Servers > primaryServer > Log**, and in the right pane click **Messaging Server > General**.
2. To display the latest log data collected in the log file if time has elapsed since you last opened the log, click **Retrieve**.
3. Click **Export** to launch the Export Trace Log Wizard.

Trace Log

The trace logs capture messaging server data for cluster level database to mobile device user activities. Using the trace logs you can trace obtain detailed information using a variety of search criteria.

- **Time** – the date and time when the current trace entry was logged on Unwired Server. The returned date and time is the Unwired Server time without time zone information.
- **Module** – the module to which the current trace entry belongs.
- **Description** – detailed trace information.
- **Level** – the trace level of the current trace entry. The possible trace level values (from high to low) are: ERROR, WARN, INFO, and DEBUG.
- **User** – the user name of the current trace entry.
- **Application Connection ID** – the application connection ID of the current trace entry.
- **Thread ID** – the thread ID when the trace entry was logged.
- **Node** – the server that created the trace entry.

Domains

Domains provide a logical partitioning of a hosting organization's environment that achieves increased flexibility and granularity of control in multitenant environments. By default, the installer creates a single domain named "default."

Administrators use different domains within the same Unwired Platform installation. Domains enable the management of application metadata within a partition, including server connections, packages, role mappings, domain logs, and security, so that changes are visible only in the specific domain.

Considerations when implementing domains in a multitenant environment include:

- Create and manage domains using Sybase Control Center from the Unwired Platform administration perspective of Sybase Control Center.
- You can support multiple customers inside the same Unwired Platform cluster.
- You can configure security specifically for individual domains by creating one or more security configurations in the cluster, and then assigning those security configurations to a domain. You can then map the security configurations to one or more packages. A user accessing the package from a device application is authenticated and authorized by the security provider associated with the package.
- Customers may require their own administrative view on their portion of the Unwired Platform-enabled mobility system. By granting domain administration access to your customers, you can allow customers to customize their deployed applications packages and perform self-administration tasks as needed.

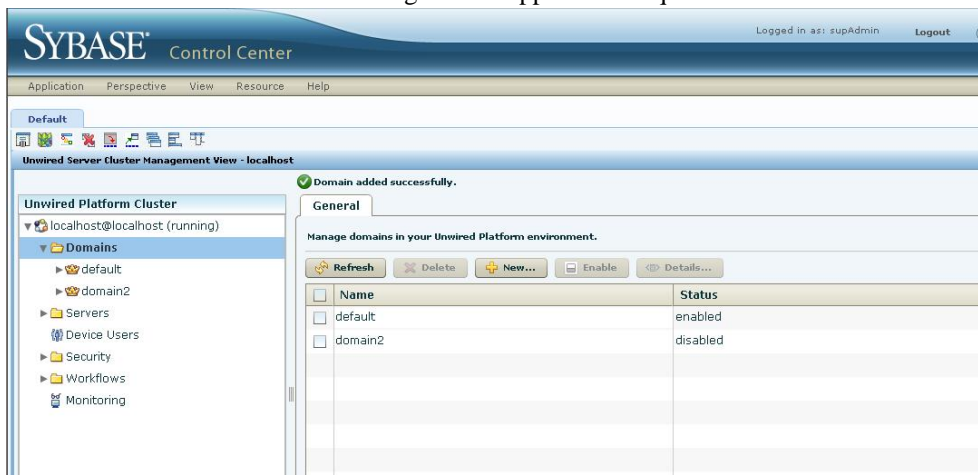
The "default" domain

The "default" domain is a special domain where critical runtime configuration artifacts exist. These artifacts include:

- An "admin" security configuration – this security configuration is mapped to the "default" domain and is used to authenticate and authorize administrative users. For this reason, administrators are not allowed to unassign the "admin" security configuration from the "default" domain.
- Cache database (CDB) data source connections – for the "default" CDB data source, users can configure the Pool Size property in the "default" domain according to their requirements. This setting allows the maximum number of open connections to the SQL Anywhere database server hosting the CDB.
- Monitor database data source connections – the customer can modify the existing monitoring data source properties according to their configuration requirements, or create a new monitoring datasource in the "default" domain.
- Domain log database data source connections – the customer can modify the existing domain log data source properties according to their configuration requirement, or create a

new domain log data source in the "default" domain. By default, the name of domain log data source is "domainlogdb".

Since these critical runtime-related artifacts are located in the "default" domain, administrators are not allowed to delete this domain. Sybase recommends creating new domains to facilitate tenants according to their application requirements.



Creating and Enabling a New Domain

Create and configure multiple domains within a single Unwired Platform installation. A domain must be enabled for application users to access the packages deployed in the domain. Enabling a domain also triggers synchronization of the domain changes to the secondary nodes in the cluster. Application users who attempt to access a disabled domain receive an error message.

Prerequisites

Create a security configuration for the domain and register the domain administrator.

Task

1. In the left navigation pane, select the **Domains** folder.
2. In the right administration pane, select the **General** tab, and click **New**.
3. In the Create Domain dialog, enter a name for the domain and click **Next**.
4. Optional. Select a security configuration for the domain by checking an option from the list of available configurations. These security configurations are then available for use in validating users accessing the packages.
5. Click **Next**.
6. Optional. Select one or more domain administrators for the domain.

7. Click **Finish**.

The new domain appears in the **General** tab.

8. Click the box adjacent to the domain name, click **Enable**, then click **Yes** to confirm.

Deleting a Domain

Remove a domain and its contents from the cluster when you no longer require the partition.

When a domain is deleted, all referenced artifacts, such as domain administrators and security configurations, are retained. However, all contained artifacts, including packages, subscription templates, device subscriptions, MBO and operation historical data, package-level role mapping, cache group settings, server connections, and domain-level role mappings for security configurations independent of any other domain, are also deleted.

To preserve a deployed package before deleting a domain, export the package to an archive file.

Note: You cannot delete the "default" domain since it contains critical runtime-related artifacts.

1. In the left navigation pane, select **Domains**.
2. In the right administration pane, click the **General** tab and select the domain you want to delete.
3. Click **Delete**.
4. In the confirmation dialog, click **Yes**.

Registering a Domain Administrator User

A platform administrator can add domain administrators, so these users can administer domains to which they are assigned. This process registers an administrator with the cluster, so the user can be assigned as an administrator for a domain.

Prerequisites

Create the user entry and map the physical role to the SUP Domain Administrator logical role in the security provider repository used to authenticate administrators in Sybase Control Center (SCC).

Task

1. In the left navigation pane, click the **Security** node.
2. In the right administration pane, click the **Domain Administrators** tab and click **New**.
3. To configure user properties for the administrator, enter:
 - **Login name** – the user name assigned to the administrator. For example, if you are using LDAP to authenticate administrators, the UID is typically used as the login name.

Administer

- (Optional) **Company name** – the name of the organization the administrator belongs to. Sybase recommends you supply this information if you are setting up Unwired Platform in a hosted environment and using domains to distinguish between different hosted solutions for different organizations.
 - (Optional) **First name** – the administrator's first name. The first name must match the one assigned to the login name in the security repository.
 - (Optional) **Last name** – the administrator's last name. The last name must match the one assigned to the login name in the security repository.
4. Click **OK** to register the administrator.
The domain administrator can now log in with his or her user login credentials (user name and password).

Next

Assign the domain administrator role to this user.

Assigning Domain Administrators to a Domain

Assign domain administration privileges to a domain administrator. You must be a platform administrator to assign and unassign domain administrators.

Prerequisites

Ensure the user is already registered as a domain administrator in the Domain Administrators tab.

Task

1. In the left navigation pane, expand the **Domains** folder, and select the domain for which to assign domain administration privileges.
2. Select the domain-level **Security** folder.
3. In the right administration pane, select the **Domain Administrators** tab, and click **Assign**.
4. Select one or more administrator users to assign to the domain by checking the box adjacent to the user name.
5. Click **OK**.
A message appears above the right administration pane menu indicating the success or failure of the assignment. If successful, the new domain administrator appears in the list of users.

Viewing Applications for a Domain

View applications registered for a specific domain.

1. In the left navigation pane, expand the **Domains** folder, and select a domain.

2. Within the domain, select **Applications**.
3. In the right pane, select the domain-level **Applications** tab.
4. Click **Refresh** to view a list of Applications IDs, and their display names and descriptions.
5. Alternatively, search for one or more application IDs.
 - a) Provide the search criteria for **Application ID** by adding a search string.
 - b) Click **Go**.
All the applications that match the search criteria provided for the selected domain are populated in the table.

Viewing Application Connections for a Domain

View application connection information for a specific domain. Optionally select the relevant columns of information to display.

Note: The association of application connections to a domain is based on the "domain" setting value in the application connections. Therefore, when registering application connections, the application template must be registered either using a template, where the domain value is appropriately configured, or in the registration wizard of the Sybase Control Center user interface.

1. In the left navigation pane, expand the **Domains** folder, and select a domain.
2. Within the domain, select **Applications**.
3. In the right pane, select the domain-level **Application Connections** tab.
4. Click **Refresh** to view a list of Users.
5. Alternatively, search for one or more users.
 - a) Provide the search criteria for **Users** by adding a search string.
 - b) Click **Go**.
All the users that match the search criteria provided for the selected domain are populated in the table.
6. (Optional) Select the columns to display (all columns, or specific columns) from the drop-down list.

Scheduling Domain-Level Cleanup

Periodically clean up accumulated data maintenance items in cache that are no longer needed.

You can automate domain-level cleanup based on a configured schedule for specific cleanup categories.

Running the cleanup options uses system resources, so Sybase recommends that you schedule these tasks when system load is lightest. Optionally you can run the cleanup tasks manually.

Administer

1. In the Sybase Control Center left navigation pane, expand the **Domains** tab and select a domain.
2. In the right pane, select the **Scheduled Task** tab.
3. Under Task, select one of the options you want to schedule, and then select **Properties** to set up its automatic schedule:

Option	Description
Subscription Cleanup	Removes subscriptions that are no longer referenced by any active users. <ul style="list-style-type: none">• Replication-based synchronization – removes subscriptions not used since the last synchronization.• Message-based synchronization – removes subscriptions if Unwired Server has not received a synchronization message since the given date.
Error History Cleanup	Removes historical data on MBO data refresh and operation replay failures, which result from system or application failures. System failures may include network problems, credential issues, and back-end system failure. Application failures may include invalid values, and non-unique data. <hr/> Note: Only error messages are removed. <hr/>
Client Log Cleanup	Removes client log records that have already been synchronized to the device, or are no longer associated with active users.
Synchronization Cache Cleanup	Removes logically deleted rows in the cache that are older than the oldest synchronization time on record in the system. Synchronization activity for all clients establish the oldest synchronization time. This cleanup task also removes unused or stale partitions.

4. Select **Enable**. Schedules run until you disable them, or they expire.

Scheduling Cleanup Options

The SUP Administrator or SUP Domain Administrator schedules domain-level data maintenance cleanup.

Set up an automatic schedule for database cleanup:

1. In the left pane, select the cluster, then the domain.
2. In the right pane, select the **Scheduled Tasks** tab.
3. Select one of the cleanup options:

Option	Description
Subscription Cleanup	<p>Removes subscriptions that are no longer referenced by any active users.</p> <ul style="list-style-type: none"> • Replication-based synchronization – removes subscriptions not used since the last synchronization. • Message-based synchronization – removes subscriptions if Unwired Server has not received a synchronization message since the given date.
Error History Cleanup	<p>Removes historical data on MBO data refresh and operation replay failures, which result from system or application failures. System failures may include network problems, credential issues, and back-end system failure. Application failures may include invalid values, and non-unique data.</p> <hr/> <p>Note: Only error messages are removed.</p>
Client Log Cleanup	<p>Removes client log records that have already been synchronized to the device, or are no longer associated with active users.</p>
Synchronization Cache Cleanup	<p>Removes logically deleted rows in the cache that are older than the oldest synchronization time on record in the system. Synchronization activity for all clients establish the oldest synchronization time. This cleanup task also removes unused or stale partitions.</p>

4. Click **Properties**.
5. In the Task Properties dialog, select the **Schedule** tab, and set the appropriate options:
 - **Schedule repeat** – select how often the schedule should run. Options are **hourly**, **daily**, **custom**, and **never**.
 - If you select **hourly** or **daily**, specify:
 - **Start date** – select the date and time the automated cleanup should begin. Use the calendar picker, and 24-hour time selector.
 - **End date** – select the date and time the automated cleanup should end.
 - **Days of the week** – select each day the automated cleanup schedule should run.
 - If you select **custom**, you can specify the interval granularity by seconds, minutes, or hours, as well as other date and time parameters.
 - If you select **never**, no scheduling options are available.
6. In the Task Properties dialog, select the **Options** tab, set the number of inactive days for which to purge.

Note: This step is unnecessary for Synchronization Cache Cleanup.

7. Click **OK** to save the schedule properties and purge options.

Enabling Domain Cleanup

The SUP Administrator or SUP Domain Administrator must enable the schedule as a separate task.

You can set up the schedule, and enable it at a later time. Once enabled, the cleanup runs automatically until is changed, disabled, or expires. You can check the current enabled or disabled status on the **Scheduled Tasks** tab.

1. In the left pane, select the cluster, then the domain.
2. In the right pane, select the **Scheduled Task** tab.
3. Select one of the cleanup options, and verify the value in the Status column is set to **disabled**.
4. On the **Scheduled Task** tab, click **Enable**.
5. Click **OK** to confirm. The value in the Status column changes to **enabled**. The cleanup schedule runs automatically for the selected option.

Disabling Domain Cleanup

The SUP Administrator or SUP Domain Administrator can disable, or reenable, a scheduled cleanup option at any time.

If you disable the cleanup option while it is running, the current process continues. Future action is disabled, unless you reenable the option.

1. In the left pane, select the cluster, then the domain.
2. In the right pane, select the **Scheduled Task** tab.
3. Select one of the cleanup options, and verify the value in the Status column is set to **enabled**.
4. On the **Schedule** tab, click **Disable**.
5. Click **OK** to confirm. The value in the Status column changes to **disabled**.

Running Manual Purge by Domain

At any time the SUP Administrator or SUP Domain Administrator can manually run cleanup options. The processes run asynchronously on Unwired Server using the current settings.

As much as reasonable, use manual purge when system load is light.

1. In the left pane, select the cluster, then the domain.
2. In the right pane, select the **Scheduled Tasks** tab.
3. Select one of the cleanup options.
4. Click **Run Now**, then optionally specify the number of days for which to preserve data. Artifacts that fall outside of the time period are purged.

5. Click **OK** to confirm. The request is sent immediately, and the task runs asynchronously on Unwired Server.

Domain Logs

The domain log enables an administrator to monitor application activities throughout the system. Detailed views of application activities are available by subsystem. The administrator can review activities in a specific subsystem log view, view correlated data in multiple subsystems, or view a unified log across all subsystems. The administrator must enable logging, and then use log filters to view data of interest.

By default, only error messages are recorded in each domain's log. To enable domain logging, you must create a log profile. See *Creating and Enabling Log Profiles* in *Sybase Control Center* online help.

Enabling Application Logging

Enable domain-level logging to help you trace and monitor application activities, and review the resulting logs for troubleshooting.

Creating and Enabling Domain Logging

Create logging profile definitions and enable the log profile.

1. In the left navigation pane of Sybase Control Center, select **Domains**.
2. Under the domain node, select **Log**.
3. In the right administration pane, select the **Settings** tab.
4. Click **New**.
5. In the Profile Definition dialog, enter a **Name** and **Description** for the log profile.
6. Add the necessary profile definitions.
7. Select **Enable after creation**.
8. Click **OK**.

Note: To ensure the domain logs are populated immediately after enabling the log profile, do the following:

1. Under the **Settings** tab, click **Configuration**.
 2. Check **Enable flush threshold**.
-

Creating the Profile Definition

Create profile definitions belonging to multiple categories.

You can add profile definitions by selecting applications, security configurations, users, connections, applications connections or payloads of your choice.

Adding or Removing Applications to the Profile

Add applications to the log profile that are currently deployed in the selected domain.

1. In the Profile Definition dialog, select **Package related**.
2. Select **Applications**, then click the button to add application to the profile.
The Applications dialog is displayed with the list of applications currently deployed in this domain.
3. To search for the application you want to add to the profile, select the search criteria from the **Search** drop-down list and enter a value for this criteria.
4. Click **Go**.
5. You can do any of the following:
 - To add an application to a new or existing profile, select the check-box adjacent to the application entry in the list.
 - To remove an application from the profile, uncheck the check-box adjacent to the application entry in the list.
6. Click **OK**.

Adding or Removing Packages to the Profile

Add one or more packages to the profile.

1. In the Profile Definition dialog, select **Package related**.
2. Select **Packages**, then click the button to add packages to the profile.
3. Select one or more packages to include.
4. Click **OK**.

Note: To remove a selection, click **View selection**, select the item, then click **Remove**.

Adding or Removing MBOs to the Profile

Add MBOs from one or more packages to the profile.

1. In the Profile Definition dialog, select **Package related**.
2. Select **MBOs**, then click the button to add MBOs to the profile.
3. Navigate to a package in the left pane.
4. Select the MBO in the right pane.
5. Identify another MBO, or click **OK**.

Note: To remove a selection, click **View selection**, select the item, then click **Remove**.

Adding or Removing Operations to the Profile

Add operations to the log profile for MBOs that are currently deployed in the selected domain. You can select operations for one or more MBOs.

1. In the Profile Definition dialog, select **Package related**.
2. Select **Operations**, then click the button to add operations to the profile.
3. Navigate to a package in the left pane, and select an MBO.
4. Identify the create, delete, or update operations to include for the MBO in the right pane.
5. Select another MBO and its operations, or click **OK**.

Note: To remove a selection, click **View selection**, select the item, then click **Remove**.

Adding or Removing Security Configurations or Users to the Profile

Add one or more domain security configurations to the profile.

1. In the Profile Definition dialog, select **Security related**.
2. Select **Security configuration**, then click the button to add packages to the profile.
3. Select one or more security configuration to include.
4. Click **OK**.

Note: To remove a selection, click **View selection**, select the item, then click **Remove**.

Adding or Removing Package Users to the Profile

Add one or more domain package users to the profile to include security configurations.

1. In the Profile Definition dialog, select **Security related**.
2. Select **Users**, then click the button to add packages to the profile.
3. Select one or more user name to include.
4. Click **OK**.

Note: To remove a selection, click **View selection**, select the item, then click **Remove**.

Adding or Removing Connections to the Profile

Add connections of a particular connection type to the profile.

1. In the Profile Definition dialog, select **Connections**.
2. Click the button to add connections to the profile.
The Connections dialog is displayed with the list of applications currently deployed in this domain.
3. You can do any of the following:

- To add a connection to a new or existing profile, select the check-box adjacent to the **Connection Type** entry in the list and select the connection names.

Note: To add connections for proxy services, select the **Proxy** connection type.

- To remove an application from the profile, select the **View selection** check-box. Select the connection type and click **Remove**.

4. Click **OK**.

Adding or Removing Application Connections to the Profile

Add application connections to the profile. This enables you to list current application connections to Unwired Server.

1. In Profile Definition, select **Application connections**.
2. Click the button to add application connections to the profile.
3. In Application Connections, select Application Connections, and the data columns to include
4. Click **OK**.

Adding or Removing Payloads to the Profile

Add a subsystem to the profile for payload logging. This enables you to identify one or more specific subsystems where payload data will also be included with the logged activity. If specified, payload is enabled for the selected subsystems.

The payload corresponds to the information of one request serviced by the SUP Server. Example: For an administrator to keep track of granular details such as request headers sent to Gateway through the proxy server, payload is enabled for a profile.

1. In the Profile Definition dialog, select **Payloads**.
2. Click the button to add payloads to the profile.
The Payloads dialog is displayed.
3. You can do any of the following:
 - Select the check-box adjacent to the subsystem you want to add to the profile.
 - Uncheck the check-box that you want to remove from the profile.
4. Click **OK**.

Enabling the Created Profile

Enable the profile you have created to monitor the log profile definitions.

1. In the Profile Definition dialog, select **Enable after creation** to enable the logging profile once you have created it.
2. Click **OK**.

You can alternatively enable the log profile by doing the following:

1. In the **Settings** tab, select the log profile you have created.
2. Click **Enable**.
3. Click **OK** on the confirmation dialog.

Enabling and Configuring Domain Logging

Configure auto-purge, flush threshold, and flush batch size settings to determine how long domain log data is retained, how frequently it is written to database from server nodes, and set a domain log database connection to configure where domain log data is stored.

1. In the left navigation pane of Sybase Control Center, select **Domains**.
2. Under the domain node, select **Log**.
3. In the right administration pane, select the **Settings** tab. These settings are used for all domains.
4. Click **Configuration**.
5. Configure auto purge settings.
Auto purge clears obsolete data from the database once it reaches the specified threshold.
 - a) Select **Enable auto purge configuration** to activate auto purge functionality.
 - b) Enter the length of time (in days) to retain monitoring data before it is purged.
6. Configure flush threshold settings:
The flush threshold indicates how often data is flushed from memory to the database. This allows you to specify the size of the data saved in memory before it is cleared. Alternately, if you do not enable a flush threshold, data is immediately written to the domain log database as it is captured.
 - a) Select **Enable flush threshold configuration** to activate flush threshold functionality.

Note: Enabling flush configuration is a good practice for performance considerations. Be aware there may be a consequent delay in viewing data, until data is stored in the database.

 - b) Select one of:
 - **Number of rows** – domain log data that surpasses the specified number of rows is flushed from memory. Enter the desired number of rows adjacent to **Rows**. Disabled by default.
 - **Time interval** – domain log data older than the specified time interval is flushed from memory. Enter the desired duration adjacent to **Minutes**. The default is 5.
 - **Either rows or time interval** – domain log data is flushed from memory according to whichever value is reached first: either the specified number of rows or the specified time interval. Enter the desired rows and duration adjacent to **Rows** and **Minutes**, respectively.
7. If you enabled a flush threshold, enter a **Flush batch row size** by specifying the size of each batch of data sent to the domain log database. The row size must be a positive integer. The batch size divides flushed data into smaller segments, rather than saving all data together according to the flush threshold parameters. For example, if you set the flush

threshold to 100 rows and the flush batch row size to 50, once 100 rows are collected in the console, the save process executes twice; data is flushed into the database in two batches of 50 rows. If the flush threshold is not enabled, the flush batch row size is implicitly 1.

Note: By default, the domain log database flushes data every 5 minutes. Alternatively, you can flush data immediately by removing or decreasing the default values, but doing so impacts performance.

8. Optional. To change the data source, select an available database from the **Domain log database endpoint** drop down list.

Available databases are those with a JDBC server connection type (SQL Anywhere) created in the default domain. To create a new database, a platform administrator must set up a database by running the appropriate configuration scripts and creating a server connection for the database in the default domain. The database server connection then appears as an option in the Domain Log Database Endpoint drop down list.

9. Click **OK**.

Reviewing Domain Log Data

An administrator reviews logged data by creating log filters. The filters enable you to retrieve data logged for a specific thread, application, user, connection, among other options.

You can retrieve log data without using any filters, however, when there is large number of activities being logged, it may be advisable to filter the results to a more manageable size by specifying search conditions in the log filter (user, application, or thread-id).

You can combine multiple log filters that are common with sub-system specific filters when viewing in a sub-system view, and combine multiple sub-system filters in the ALL tab to retrieve the data of interest.

Supported Log Subsystems

Log subsystems provide categories that enable you to filter and correlate application data at a more granular level. Understanding these subsystems enables you to formulate more specific filters for tracking application activities throughout the system.

Subsystem	Description
All	Provides a unified view of all subsystems, enabling you to look for activities, trends, and symptoms across multiple subsystems.
Synchronization	Provides a view of synchronization activities. Within this subsystem, additional categories include data synchronization, operation replay, subscription, result checker, cache refresh, and data services (DS) interface.
Device Notification	Provides a view of device notification activities.

Subsystem	Description
DCN	Provides a view of data change notification (DCN) activities. Within this subsystem, additional categories include general DCN, and workflow DCN.
Security	Provides a view of security-related activities.
Error	Provides a view of errors.
Connection	Provides a view of connection activities. Within this subsystem, additional categories include DOE, JDBC, RES, SAP®, and SOAP connections.
Proxy	Provides a view of Online Data Proxy connection-related activities. Within this subsystem, categories include request response, and push.

Setting Up a Pool of Log Filters

Set up a pool of log filters to filter out unwanted application activities, and provide a view of specific activities. Use Sybase Control Center to create and manage your filters.

Reusable Log Filters

Create reusable log filters that you can use as a base. One strategy is to create a base log filter for each of the supported log subsystems, and for significant categories within subsystems. Another strategy is to create common log filters (useful across subsystems) on specific criteria, such as thread ID, user, package, and so forth.

You can modify these base log filters as needed for more specific searches, or clone the log filter and modify it for a specific search.

Creating Log Filters

Filter the log data by creating filters across subsystems that define the appropriate search criteria.

1. In the left navigation pane of the Sybase Control Center, select the **Domains** node.
2. Select the domain node and select the **Log** node.
3. In the right administration pane, select the **General** tab.
4. Select + to add a filter definition to a subsystem.
5. In the Filter Definition dialog, enter the **Name** and **Description** of the filter.
6. Select the **Sub System**.
7. Select the filter criteria and assign values to the criteria selected. You can use the logical operations to compose the criteria.

Note: You use the 'AND' logical operator to highlight filter relations belonging to the same subsystem. Filter definitions among multiple subsystems use the 'OR' logical operator.

8. Click **OK**.

Deleting Filters

Delete the filters created for sub systems

1. In the left navigation pane of the Sybase Control Center, select the **Domains** node.
2. Select the domain node and select the **Log** node.
3. In the right administration pane, select the **General** tab.
4. Select the filter from the list.
5. Click **Delete**.

Updating Filters

Update filters as needed to fine tune log file filtering.

1. In the left navigation pane of the Sybase Control Center, select the **Domains** node.
2. Select the domain node and select the **Log** node.
3. In the right administration pane, select the **General** tab.
4. Select the filter from the list.
5. Click the properties icon to review or modify the filter.

Note: Alternatively, click the clone icon to clone the filter, then proceed to modify it.

6. In Filter Definition, modify the description, and set up the filter criteria.
7. Click **OK**.

Retrieving Unified View Logs

Retrieves logging data across the domain to provide a unified view.

1. Display the **General** tab for Domain Logs.
In the navigation pane, click **Domain** > <domainName> > **Log**, then select **General** from the administration pane.
2. Select the **All** tab.
3. To filter the display, select **Show filter** and either:
 - Use an existing filter by checking the box adjacent to the filter name.
 - Create a new filter by clicking + and choosing a starting date and time, and ending date and time, and setting a **Time Order** (ascending or descending).

To customize the display, select the view type (grid or text), or click >> to show only selected columns. To export the filtered results, click **Export** and select an appropriate output destination.

4. Click **Retrieve** to retrieve the logs.

The table is populated with the list of logs.

Retrieving Synchronization Logs

Retrieves logging data related to different aspects of data synchronization, including data, subscriptions, operations, result checker, cache refresh and the data service and Unwired Server interface. Using data in these logs and the correlation tool, you can follow the data path between the enterprise information system (EIS), Unwired Server, cache database, and user application connection.

1. Display the **General** tab for Domain Logs.

In the navigation pane, click **Domain** > <domainName> > **Log**, then select **General** from the administration pane.

2. Select the **Synchronization** tab.

3. Select a synchronization subsystem:

- Data Sync – view data synchronization details.
- Operation Replay – view MBO operational replay details.
- Subscription – view subscription details.
- Result Checker – view result checker details.
- Cache Refresh – view cache database interaction details.
- DS Interface – view requests entering data services that result in interaction with cache database and EIS.

4. To filter the display, select **Show filter** and either:

- Use an existing filter by checking the box adjacent to the filter name.
- Create a new filter by clicking + and choosing a starting date and time, and ending date and time, and setting a **Time Order** (ascending or descending).

To customize the display, select the view type (grid or text), or click >> to show only selected columns. To export the filtered results, click **Export** and select an appropriate output destination.

5. (Optional) Select the columns to display from the drop down list, such as Application ID, Application Connection Id, and User.
6. Click **Retrieve** to retrieve the logs.
7. (Optional) Select a specific row to view additional columns in the detail area.

Retrieving Device Notification Logs

Retrieves logging data for server-initiated synchronization notifications between Unwired Server and devices.

1. Display the **General** tab for Domain Logs.

In the navigation pane, click **Domain** > <domainName> > **Log**, then select **General** from the administration pane.

2. Select the **Device Notification** tab.
3. To filter the display, select **Show filter** and either:
 - Use an existing filter by checking the box adjacent to the filter name.
 - Create a new filter by clicking + and choosing a starting date and time, and ending date and time, and setting a **Time Order** (ascending or descending).

To customize the display, select the view type (grid or text), or click >> to show only selected columns. To export the filtered results, click **Export** and select an appropriate output destination.

4. (Optional) Select the columns to display from the drop down list, such as Application ID, Application Connection Id, and User.
5. Click **Retrieve** to retrieve the log data.
6. (Optional) Select a specific row to view additional columns in the detail area.

Retrieving Data Change Notification Logs

Retrieves logging data for data change notifications (DCN) between an enterprise information system (EIS) and an MBO package, for general and workflow DCN.

1. Display the **General** tab for Domain Logs.

In the navigation pane, click **Domain** > <domainName> > **Log**, then select **General** from the administration pane.

2. Select the **DCN** tab.
3. Select the DCN type:
 - General DCN
 - Workflow DCN
4. To filter the display, select **Show filter** and either:
 - Use an existing filter by checking the box adjacent to the filter name.
 - Create a new filter by clicking + and choosing a starting date and time, and ending date and time, and setting a **Time Order** (ascending or descending).

To customize the display, select the view type (grid or text), or click >> to show only selected columns. To export the filtered results, click **Export** and select an appropriate output destination.

5. (Optional) Select the columns to display from the drop down list, such as Application ID, Application Connection Id, and User.
6. Click **Retrieve** to retrieve the logs.
7. (Optional) Select a specific row to view additional columns in the detail area.

Retrieving Security Logs

Retrieves security details for specific applications.

1. Display the **General** tab for Domain Logs.

In the navigation pane, click **Domain** > <domainName> > **Log**, then select **General** from the administration pane.

2. Select the **Security** tab.

3. To filter the display, select **Show filter** and either:

- Use an existing filter by checking the box adjacent to the filter name.
- Create a new filter by clicking + and choosing a starting date and time, and ending date and time, and setting a **Time Order** (ascending or descending).

To customize the display, select the view type (grid or text), or click >> to show only selected columns. To export the filtered results, click **Export** and select an appropriate output destination.

4. (Optional) Select the columns to display from the drop down list, such as Application ID, Application Connection Id, and User.
5. Click **Retrieve** to retrieve the logs.

Retrieving Errors Logs

Retrieves logging data for domain errors. Note that error logging is always on, and any error that occurs for any application activity is logged.

1. Display the **General** tab for Domain Logs.

In the navigation pane, click **Domain** > <domainName> > **Log**, then select **General** from the administration pane.

2. Select the **Error** tab.

3. To filter the display, select **Show filter** and either:

- Use an existing filter by checking the box adjacent to the filter name.
- Create a new filter by clicking + and choosing a starting date and time, and ending date and time, and setting a **Time Order** (ascending or descending).

To customize the display, select the view type (grid or text), or click >> to show only selected columns. To export the filtered results, click **Export** and select an appropriate output destination.

4. (Optional) Select the columns to display from the drop down list, such as Application ID, Application Connection Id, and User.
5. Click **Retrieve** to retrieve the logs.

Retrieving Logs for Connections

Retrieves logging data for domain connections for specific connection types to backend data sources, including DOE, JDBC, REST, SAP, and SOAP if enabled.

1. Display the **General** tab for Domain Logs.

In the navigation pane, click **Domain** > <domainName> > **Log**, then select **General** from the administration pane.

2. Select the **Connection** tab.
3. Select a connection type, such as DOE, JDBC, REST, SAP, or SOAP.
4. To filter the display, select **Show filter** and either:
 - Use an existing filter by checking the box adjacent to the filter name.
 - Create a new filter by clicking + and choosing a starting date and time, and ending date and time, and setting a **Time Order** (ascending or descending).

To customize the display, select the view type (grid or text), or click >> to show only selected columns. To export the filtered results, click **Export** and select an appropriate output destination.

5. (Optional) Select the columns to display from the drop down list, such as Application ID, Application Connection Id, and User.
6. Click **Retrieve** to retrieve the data.
7. (Optional) Select a specific row to view additional columns in the detail area.

Retrieving Proxy Request-Response Logs

Retrieves the log data for all requests and responses made from the Proxy server.

Once you have traced a connection under the Applications node, you can retrieve the logs under the Domains node.

1. Display the **General** tab for Domain Logs.

In the navigation pane, click **Domain** > <domainName> > **Log**, then select **General** from the administration pane.

2. Select the **Proxy** tab and then select **Request Response**.
3. To filter the display, select **Show filter** and either:
 - Use an existing filter by checking the box adjacent to the filter name.
 - Create a new filter by clicking + and choosing a starting date and time, and ending date and time, and setting a **Time Order** (ascending or descending).

To customize the display, select the view type (grid or text), or click >> to show only selected columns. To export the filtered results, click **Export** and select an appropriate output destination.

4. (Optional) Select the columns to display from the drop down list, such as Application ID, Application Connection Id, and User.
5. Click **Retrieve** to retrieve the log data.
6. (Optional) Select a specific row to view additional columns in the detail area.

Retrieving Proxy Push Logs

Retrieves the log data for all push notifications from the Proxy server.

Once you have traced a connection under the Applications node, you can retrieve the logs under the Domains node.

1. Display the **General** tab for Domain Logs.
In the navigation pane, click **Domain** > <domainName> > **Log**, then select **General** from the administration pane.
2. Select the **Proxy** tab and then select **Push**.
3. To filter the display, select **Show filter** and either:
 - Use an existing filter by checking the box adjacent to the filter name.
 - Create a new filter by clicking + and choosing a starting date and time, and ending date and time, and setting a **Time Order** (ascending or descending).

To customize the display, select the view type (grid or text), or click >> to show only selected columns. To export the filtered results, click **Export** and select an appropriate output destination.

4. (Optional) Select the columns to display from the drop down list, such as Application ID, Application Connection Id, and User.
5. Click **Retrieve** to retrieve the data.
6. (Optional) Select a specific row to view additional columns in the detail area.

Correlating Log Data Across Subsystems

Correlation mode enables you to retrieve domain log data in multiple subsystems, using the same search condition. The same condition is combined by common type filters and time range. This provides a tool for correlating activity across subsystems, useful for analyzing and troubleshooting.

For example, you could create a common filter for Application ID (such as Application ID = appid); select the ProxyRequestResponse tab and enter the filter information; retrieve the data; then select Correlated mode, and switch to another tab such as ProxyPush.

1. In the left navigation pane of Sybase Control Center, select **Domain**.
2. Under the domain node, select **Log**.
3. In the right administration pane, select the **General** tab.
4. Select the subsystem tab, such as **Synchronization**.

5. To select or set up a special filter, select **Show filter**.
6. To filter based on the date and time, enter a **Start date, End date, Start time, End time**.
7. (Optional) Select the columns to display from the drop down list, such as Application ID, Application Connection Id, User, and Thread.
8. Click **Retrieve** to retrieve the logs.
The table is populated with log data.
9. Select **Correlated mode** and select the common type of log filters to use.
10. Switch to another subsystem tabs. The data is refreshed using the same criteria from the common type of log filters and time range specified.
11. Use the data to trace application activity across subsystems.

Exporting Log Data

Export data to a file for archive, or to analyze and troubleshoot problems.

Note:

- Depending on the amount of data being exported, the log export can take a long time. Sybase recommends using log filters and time ranges to filter out and export specific log entries of interest.
 - You can also use the management API if there is a need to export domain log contents if the data set is large, or to refrain from blocking the user interface.
-

1. In the left navigation pane of Sybase Control Center, select **Domain**.
2. Under the domain node, select **Log**.
3. In the right administration pane, select the **General** tab.
4. Select the subsystem tab, such as **Synchronization**.
5. To select or set up a special filter, select **Show filter**.
6. To filter based on the date and time, enter a **Start date, End date, Start time, End time**.
7. Click **Retrieve** to retrieve the logs.
The table is populated with log data.
8. Click **Export** and specify the file name and location.

Purging Domain Logs

You can manually purge the domain log.

1. In the left navigation pane of Sybase Control Center, select **Domains**.
2. Under the domain node, select **Log**.
3. In the right administration pane, select the **Settings** tab.

4. Click **Purge**.
5. Enter the date and time within which you want the data to be purged.
6. Click **OK**.

Domain Log Categories

Domain log data provides detailed statistics for all aspects of device, user, application, domain, and data synchronization related activities.

Synchronization Log

Synchronization logs include data related to different aspects of data synchronization, including data, subscriptions, operations, result checker, cache refresh and the data service and Unwired Server interface. Using data in these logs and the correlation tool, you can follow the data path between the enterprise information system (EIS), Unwired Server, cache database, and user application connection.

To find out about	See
Data synchronization transactions	Data Sync statistics
Data services requests made to the Enterprise information system (EIS)	DS Interface statistics
Cache database (CDB) activities	Cache refresh statistics
EIS error codes or failures resulting from Mobile Business Object operations against the EIS data-source	Result Checker statistics (coding required)
Moving MBO operations from a mobile device to the CDB	Operation replay statistics
Moving data between a mobile device and the CDB	Subscription statistics

Data Sync

Data Sync – basic statistics for individual data synchronizations:

- Time – the time and date stamp for the log entry.
- Application ID – the unique identifier assigned to the registered application. Values may include a number, blank, or HWC, depending on the client type.
- Application Connection ID – the unique identifier for a user application connection.
- User – the name of the user associated with the application ID.
- Stage – the current stage of processing - START or FINISH.
- Package – the name of the package to which the subscription belongs.
- MBO – the mobile business object used.

Administer

- Sync Group – the synchronization group associated with the request.
 - Thread ID – the identifier for the thread used to process the request.
 - Node ID – the server node on which the request is received.
 - Error – the error message if any.
-

Note: Additional detail columns:

- Payload
-

Operation Replay

Operation Replay – statistics for moving MBO operations (typically create, update, and delete) from the device cache to the cache database cache on Unwired Server:

- Time – the time and date stamp for the log entry.
 - Application ID – the unique identifier assigned to the registered application. Values may include a number, blank, or HWC, depending on the client type.
 - Application Connection ID – the unique identifier for a user application connection.
 - User – the name of the user associated with the application ID.
 - Stage – the current stage of processing - START or FINISH.
 - Package – the name of the package to which the subscription belongs.
 - MBO – the mobile business object used.
 - Operation – the MBO operation.
 - Thread ID – the identifier for the thread used to process the request.
 - Node ID – the server node on which the request is received.
 - Error – the error message if any.
-

Note: Additional detail columns:

- Payload
-

Subscription

Subscription – statistics for transferring data between mobile devices and the cache database on Unwired Server:

- Time – the time and date stamp for the log entry.
- Application ID – the unique identifier assigned to the registered application. Values may include a number, blank, or HWC, depending on the client type.
- Application Connection ID – the unique identifier for a user application connection.
- User – the name of the user associated with the application ID.
- Stage – the current stage of processing - START or FINISH.
- Package – the name of the package to which the subscription belongs.
- Subscription Type – the type of subscription used, including SUBSCRIBE, UNSUBSCRIBE, RECOVER, SUSPEND, and RESUME.

- Subscription ID – the identifier associated with the subscription.
- Sync Group – the synchronization group associated with the request.
- Thread ID – the identifier for the thread used to process the request.
- Node ID – the server node on which the request is received.
- Error – the error message if any.

Note: Additional detail columns:

- Payload
-

Result Checker

Result Checker – EIS error codes or failures resulting from Mobile Business Object operations against the EIS datasource (requires coding):

- Time – the time and date stamp for the log entry.
- Application ID – the unique identifier assigned to the registered application. Values may include a number, blank, or HWC, depending on the client type.
- Application Connection ID – the unique identifier for a user application connection.
- User – the name of the user associated with the application ID.
- Stage – the current stage of processing - START or FINISH.
- Package – the name of the package to which the subscription belongs.
- Class – the class used for the result checker.
- Thread ID – the identifier for the thread used to process the request.
- Node ID – the server node on which the request is received.
- Error – the error message if any.

Note: Additional detail columns:

- None
-

Cache Refresh

Cache Refresh – statistics for cache database activities:

- Time – the time and date stamp for the log entry.
- Application ID – the unique identifier assigned to the registered application. Values may include a number, blank, or HWC, depending on the client type.
- Application Connection ID – the unique identifier for a user application connection.
- User – the name of the user associated with the application ID.
- Stage – the current stage of processing - START or FINISH.
- Package – the name of the package to which the subscription belongs.
- MBO – the mobile business object used.
- Cache Group – the cache group name.

Administer

- CacheRow Count – the number of cached rows.
 - EIS Row Count – the number of rows retrieved from the enterprise information system (EIS).
 - Insert Count – the number of rows inserted in the cache.
 - Update Count – the number of rows updated in the cache.
 - Delete Count – the number of rows deleted from the cache.
 - Thread ID – the identifier for the thread used to process the request.
 - Node ID – the server node on which the request is received.
 - Error – the error message if any.
-

Note: Additional detail columns:

- Refresh Type
 - Virtual Table Name
 - Partition Key
 - Pre Update Cache Image
 - Post Update Cache Image
-

DS Interface

DS Interface – statistics for data services requests made to the Enterprise information system (EIS):

- Time – the time and date stamp for the log entry.
 - Application ID – the unique identifier assigned to the registered application. Values may include a number, blank, or HWC, depending on the client type.
 - Application Connection ID – the unique identifier for a user application connection.
 - User – the name of the user associated with the application ID.
 - Stage – the current stage of processing - START or FINISH.
 - Package – the name of the package to which the subscription belongs.
 - MBO – the mobile business object used.
 - Operation – the MBO operation.
 - Thread ID – the identifier for the thread used to process the request.
 - Node ID – the server node on which the request is received.
 - Error – the error message if any.
-

Note: Additional detail columns:

- Operation Type
 - Virtual Table Name
 - Input Attributes (payload)
 - Input Parameters (payload)
-

Device Notification Log

Device notification logs include logging data for server-initiated synchronization notifications between Unwired Server and devices.

- Time – the time and date stamp for the log entry.
- Application ID – the unique identifier assigned to the registered application. Values may include a number, blank, or HWC, depending on the client type.
- Application Connection ID – the unique identifier for a user application connection.
- User – the name of the user associated with the application ID.
- Package – the name of the package to which the subscription belongs.
- MBO – the mobile business object used.
- Sync Group – the synchronization group associated with the request.
- Thread ID – the identifier for the thread used to process the request.
- Node ID – the server node on which the request is received.
- Error – the error message if any.

Note: Additional detail columns:

- Payload
-

Data Change Notification Log

Data Change Notification (DCN) logs include logging data for data change notifications between an enterprise information system (EIS) and an MBO package, for general and workflow DCN.

General Data Change Notification

For general DCN:

- Time – the time and date stamp for the log entry.
- User – the name of the user associated with the application ID.
- Stage – the current stage of processing - START or FINISH.
- Package – the name of the package to which the subscription belongs.
- MBO – the mobile business object used.
- Thread ID – the identifier for the thread used to process the request.
- Node ID – the server node on which the request is received.
- Error – the error message if any.

Note: Additional detail columns:

- Payload
-

Workflow Data Change Notification

For workflow DCN:

Administer

- Time – the time and date stamp for the log entry.
- Workflow ID – the unique identifier associated with a workflow.
- Application Connection ID – the unique identifier for a user application connection.
- User – the name of the user associated with the application ID.
- Package – the name of the package to which the subscription belongs.
- Thread ID – the identifier for the thread used to process the request.
- Node ID – the server node on which the request is received.
- Operation – the MBO operation.
- Subject – the workflow DCN request subject line.
- From – the "From" value for the workflow DCN request.
- To – the "To" value for the workflow DCN request.
- Body – the message body for the workflow DCN request.
- Error – the error message if any.

Note: Additional detail columns:

- Payload
-

Security Log

Security logs provide security details for individual applications, application connections, and users. Logs capture authentication failures and errors, and provide supporting information that identifies request-response messaging, package and MBO details, security configuration, and the thread and node that attempted to process an authentication request.

- Time – the time and date stamp for the log entry.
- Application ID – the unique identifier assigned to the registered application. Values may include a number, blank, or HWC, depending on the client type.
- Application Connection ID – the unique identifier for a user application connection.
- User – the name of the user associated with the application ID.
- Correlation ID – the unique ID associated with every request-response message pair.
- Package – the name of the package to which the subscription belongs.
- MBO – the mobile business object used.
- Security Configuration – the associated security configuration.
- Method – the MBO operation used.
- Thread ID – the identifier for the thread used to process the request.
- Node ID – the server node on which the request is received.
- Outcome – the authentication outcome for the security check.
- Reason – the reason for authentication failure.
- Error – the error message if any.

Error Log

Errors log data includes domain-level errors.

- Time – the time and date stamp for the log entry.
- Application ID – the unique identifier assigned to the registered application. Values may include a number, blank, or HWC, depending on the client type.
- Application Connection ID – the unique identifier for a user application connection.
- User – the name of the user associated with the application ID.
- Correlation ID – the unique ID associated with every request-response message pair.
- Package – the name of the package to which the subscription belongs.
- MBO – the mobile business object used.
- Operation – the MBO operation.
- Thread ID – the identifier for the thread used to process the request.
- Node ID – the server node on which the request is received.
- Error – the error message if any.

Connection Log

Connections log data includes domain connections for specific connection types to backend data sources, including DOE, JDBC, REST, SAP, and SOAP, if enabled. Additional columns may be available in the detail pane, and payload data may be available if enabled.

DOE Connection

- Time – the time and date stamp for the log entry.
- Application ID – the unique identifier assigned to the registered application. Values may include a number, blank, or HWC, depending on the client type.
- Application Connection ID – the unique identifier for a user application connection.
- User – the name of the user associated with the application ID.
- Event Type – the DOE-C event type, such as Acknowledged, Ignored, Exclude, Resend (from client), and Status.
- Package – the name of the package to which the subscription belongs.
- MBO – the mobile business object used.
- Operation – the MBO operation.
- Connection – the managed connection used.
- Client ID – the identifier for the DOE-C client.
- Physical ID – the DOE-C generated physical identifier registered with DOE at subscription.
- Subscription ID – the DOE-C generated subscription identifier registered with DOE at subscription.
- Logical Device ID – the DOE-C logical device identifier, generated by DOE and provided to DOE-C upon successful subscription.

Administer

- Message Direction – the DOE-C message direction, either client to Unwired Server, or Unwired Server to client.
 - Thread ID – the identifier for the thread used to process the request.
 - Node ID – the server node on which the request is received.
 - Error – the error message if any.
-

Note: Payload and detail columns:

- Endpoint Name (payload)
 - JSON Message Content (payload)
 - XML Message Content (payload)
 - DOE server message ID
 - DOE client message ID
 - DOE-C server message ID
 - DOE-C client message ID
 - DOE-C method name
 - DOE-C action name
 - Push to
 - Address
 - Log
 - Extract Window
 - PBI
-

JDBC Connection

- Time – the time and date stamp for the log entry.
 - Application ID – the unique identifier assigned to the registered application. Values may include a number, blank, or HWC, depending on the client type.
 - Application Connection ID – the unique identifier for a user application connection.
 - User – the name of the user associated with the application ID.
 - Stage – the current stage of processing - START or FINISH.
 - Package – the name of the package to which the subscription belongs.
 - MBO – the mobile business object used.
 - Operation – the MBO operation.
 - Connection – the managed connection used.
 - Thread ID – the identifier for the thread used to process the request.
 - Node ID – the server node on which the request is received.
 - Error – the error message if any.
-

Note: Payload and detail columns:

- Input Parameters (payload)

- Query (payload)
 - Endpoint Name
 - Database Product Name
 - Database Product Version
 - Driver Name
 - Driver Version
 - Database User Name
-

REST Connection

- Time – the time and date stamp for the log entry.
 - Application ID – the unique identifier assigned to the registered application. Values may include a number, blank, or HWC, depending on the client type.
 - Application Connection ID – the unique identifier for a user application connection.
 - User – the name of the user associated with the application ID.
 - Stage – the current stage of processing - START or FINISH.
 - Package – the name of the package to which the subscription belongs.
 - MBO – the mobile business object used.
 - Operation – the MBO operation.
 - Connection – the managed connection used.
 - URL – the URL associated with the managed connection.
 - Action – the GET, POST, PUT, or DELETE action.
 - Response Status – the response status code for the invocation.
 - Thread ID – the identifier for the thread used to process the request.
 - Node ID – the server node on which the request is received.
 - Error – the error message if any.
-

Note: Payload and detail columns:

- Response (payload)
 - Endpoint Name
 - HTTP Header Parameters
-

SAP Connection

- Time – the time and date stamp for the log entry.
- Application ID – the unique identifier assigned to the registered application. Values may include a number, blank, or HWC, depending on the client type.
- Application Connection ID – the unique identifier for a user application connection.
- User – the name of the user associated with the application ID.
- Stage – the current stage of processing - START or FINISH.

Administer

- Package – the name of the package to which the subscription belongs.
 - MBO – the mobile business object used.
 - Operation – the MBO operation.
 - BAPI – the SAP BAPI used as the data source.
 - Connection – the managed connection used.
 - Properties – the list of name:value pairs.
 - Thread ID – the identifier for the thread used to process the request.
 - Node ID – the server node on which the request is received.
 - Error – the error message if any.
-

Note: Payload and detail columns:

- Parameters (payload)
 - Endpoint Name
 - SAP Host
 - SAP User
-

SOAP Connection

- Time – the time and date stamp for the log entry.
 - Application ID – the unique identifier assigned to the registered application. Values may include a number, blank, or HWC, depending on the client type.
 - Application Connection ID – the unique identifier for a user application connection.
 - User – the name of the user associated with the application ID.
 - Stage – the current stage of processing - START or FINISH.
 - Package – the name of the package to which the subscription belongs.
 - MBO – the mobile business object used.
 - Operation – the MBO operation.
 - Connection – the managed connection used.
 - Service Address – the service address URL.
 - Action – the SOAP action.
 - Thread ID – the identifier for the thread used to process the request.
 - Node ID – the server node on which the request is received.
 - Error – the error message if any.
-

Note: Payload and detail columns:

- Request (payload)
 - Response (payload)
 - Endpoint Name
 - Connection Timeout
 - Authentication Type
-

Proxy Request-Response Log

Proxy Request-Response log data includes data for all requests and responses made from the Proxy server.

- Time – the time and date stamp for the log entry.
- Application ID – the unique identifier assigned to the registered application. Values may include a number, blank, or HWC, depending on the client type.
- Application Connection ID – the unique identifier for a user application connection.
- User – the name of the user associated with the application ID.
- Source - the source of the log if its from the server or client.
- Correlation ID - the unique id associated with every request-response message pair.
- Request Type - the request type of the message.
- Request URL - the Gateway URL.
- HTTP Endpoint - the Gateway URL.
- Log Level - not relevant.
- Thread ID – the identifier for the thread used to process the request.
- Node ID – the server node on which the request is received.
- Error – the error message if any.

Note: Additional detail columns:

- Post Data
 - Request Header Fields
 - Response Body
 - Response Header Fields
-

Proxy Push Log

Proxy push logs include log data for all push notifications from the Proxy server.

- Time – the time and date stamp for the log entry.
- Application ID – the unique identifier assigned to the registered application. Values may include a number, blank, or HWC, depending on the client type.
- Application Connection ID – the unique identifier for a user application connection.
- User – the name of the user associated with the application ID.
- Source - the source of the log if its from the server or client.
- Correlation ID - the unique id associated with every request-response message pair.
- URN - not relevant.
- Log Level - not relevant.
- Thread ID – the identifier for the thread used to process the request.
- Node ID – the server node on which the request is received.
- Error – the error message if any.

Note: Additional detail columns:

- Message Body
-

Checking Client Application Logs

Review data about client application operations for all devices subscribed to a package in order to track errors and identify performance issues.

1. In the left navigation pane, expand the **Packages** folder and select the package you want to manage.
2. In the right administration pane, select the **Client Log** tab.
3. Review this information to monitor client application activity:
 - User – the name of the user that activates the device.
 - Application Connection ID – the unique identifier for a user application connection.
 - MBO – the mobile business object that the client is synchronizing with.
 - Operation – the operation that the client is performing.
 - Code – the result of server-side operations; either 200 (successful) or 500 (failed).
 - Level – the log level for the application; either FATAL, ERROR, WARN, INFO, DEBUG, or TRACE.
 - Timestamp – the date and time at which the operation took place.
 - Message – the log message associated with the operation.
4. Select a row from the table and click **Details** to see a detailed view of data for the selected client log event.
5. Click **Close** to return to the Client Log summary view.

Cleaning the Client Log

Clears client application log data from the SCC administration page.

1. In the left navigation pane, expand the **Packages** folder and select the package you want to manage.
2. In the right administration pane, select the **Client Log** tab.
3. Click **Clean**.
4. Enter a time frame to indicate which client log data you want to erase, and click **OK**.
The data is removed from the Client Log tab.

Connections

Connections allow Unwired Server to communicate with data sources. To facilitate the connection process, define a set of properties for each data source. Establish connections and connection pools for each domain.

A connection is required to send queries to mobile business objects, and to receive answers. The format in which data is communicated depends on the type of data source; for example,

database data sources use a result set, while Web services data sources provide XML files, and SAP data sources use tables.

Establish connections by supplying an underlying driver and a connection string. Together, the driver and string allow you to address the data source, and provide you a mechanism by which to set the appropriate user authentication credentials and connection properties that describe the connection instance. Once a connection is established, Unwired Server can open and close it as required.

Connection Pools

Unwired Server maintains database connections in a connection pool, which is a cache database connections for the cache database or any other database data source.

A connection can be reused when the database receives future requests for data, thereby improving Unwired Server performance. If all the connections are being used, and the `maxPoolSize` value you configured for a connection pool has been reached, a new connection is added to the pool. For Unwired Server, connection pools are based on an existing template created for a specific data source type.

Connection Templates

A connection template is a model or pattern used to standardize connection properties and values for a specific connection pool type so that they can be reused. A template allows you to quickly create actual connections.

Often, setting up a connection for various enterprise data sources requires each administrator to be aware of the mandatory property names and values for connecting to data sources. Once you create a template and add appropriate property names and corresponding values (for example user, password, database name, server name, and so on), you can use the template to instantiate actual connection pools with predefined property name and value pairs.

Creating Connections and Connection Templates

Create a new connection or connection template that defines the properties needed to connect to a new data source.

1. In the left navigation pane, expand the **Domains** folder, and select the domain for which you want to create a new connection.
2. Select **Connections**.
3. In the right administration pane:
 - To create a new connection – select the **Connections** tab, and click **New**.
 - To create a new connection template – select the **Templates** tab, and click **New**.
4. Enter a unique **Connection pool name** or template name.
5. Select the **Connection pool type** or template type:
 - JDBC – choose this for most database connections.

Administer

- Proxy - choose this if you are connecting to the Online Data Proxy.
 - WS – choose this if you are connecting to a Web Services (SOAP or REST) data source.
 - SAP – choose this if you are connecting to an SAP (JCO) datasource.
6. Select the appropriate template for the data source target from the **Use template** menu. By default, several templates are installed with Unwired Platform; however, a production version of Unwired Server may have a different default template list.
 7. Template default properties appear, along with any predefined values. You can customize the template, if required, by performing one of:
 - Editing existing property values – click the corresponding cell and change the value that appears.
 - Adding new properties – click the **<ADD NEW PROPERTY>** cell in the Property column and select the required property name. You can then set values for any new properties you add.

Note: In a remote server environment, if you edit the `sampledb Server Name` property, you must specify the remote IP number or server name. Using the value "localhost" causes cluster synchronization to fail.

8. Test the values you have configured by clicking **Test Connection**. If the test fails, either values you have configured are incorrect, or the data source target is unavailable. Evaluate both possibilities and try again.
9. Click **OK** to register the connection pool.
The name appears in the available connection pools table on the Connections tab.
Administrators can now use the connection pool to deploy packages.

Editing Connection Pools and Templates

Edit the properties and values assigned to connection pools and templates.

1. In the left navigation pane, expand the **Domains** folder, and select the domain for which you want to create a new connection.
2. Select **Connections**.
3. In the right administration pane:
 - To edit the properties of a connection pool, click the **Connections** tab.
 - To edit the properties of a connection pool template, click the **Templates** tab.
4. Select a connection pool or template from the list.
5. Click **Properties**.
 - a) Edit the property and value.
 - a) Click **Save** to save the changes.

Testing a Connection

Test connection properties of a data source to validate the connection values.

1. In the left navigation pane, click the **Connections** icon.
2. Select the **Connection Pool Name** you want to validate.
3. Click **Properties**.
4. Click **Test Connection**.

If the connection test is not successful, see *Connection Test Errors* in the *Troubleshooting* guide.

Deleting a Connection Pool and Template

Delete a connection pool or template.

1. In the left navigation pane, expand the **Domains** folder, and select the domain for which you want to create a new connection.
2. Select **Connections**.
3. In the right administration pane:
 - To delete a connection pool, click the **Connections** tab.
 - To delete a connection pool template, click the **Templates** tab.
4. Select the connection pool or template you want to delete.
5. Click **Delete**.

EIS Data Source Connection Properties Reference

Name and configure connection properties when you create connection pools in Sybase Control Center to enterprise information systems (EIS) .

JDBC Properties

Configure Java Database Connectivity (JDBC) connection properties.

This list of properties can be used by all datasource types. Sybase does not document native properties used only by a single driver. However, you can also use native driver properties, naming them using this syntax:

```
<driver_type> : <NativeConnPropName>=<SupportedValue>
```

Note: If Unwired Server is connecting to a database with a JDBC driver, ensure you have copied required JAR files to correct locations. See the *Installation for Runtime* guide.

Name	Description	Supported values
After Insert	Changes the value to <code>into</code> if a database requires <code>insert into</code> rather than the abbreviated <code>into</code> .	<code>into</code>
Batch Delimiter	Sets a delimiter, for example, a semicolon, that can be used to separate multiple SQL statements within a statement batch.	<code><delimiter></code>
Blob Updater	Specifies the name of a class that can be used to update database BLOB (long binary) objects when the BLOB size is greater than <code>psMaximumBlobLength</code> .	<code><class name></code> The class must implement the <code>com.sybase.djc.sql.BlobUpdater</code> interface.
Clob Updater	Specifies the name of a class that can be used to update database CLOB (long string) objects when the CLOB size is greater than <code>psMaximumClobLength</code> .	<code><class name></code> The class must implement the <code>com.sybase.djc.sql.ClobUpdater</code> interface.
Code Set	Specifies how to represent a repertoire of characters by setting the value of <code>CS_SYB_CHARSET</code> for this datasource. Used when the data in the datasource is localized. If you do not specify the correct code set, characters may be rendered incorrectly.	<code>[server]</code> If the value is <code>server</code> , the value of the current application server's <code>defaultCodeSet</code> property is used.

Name	Description	Supported values
Commit Protocol	<p>Specifies how Unwired Server handles connections for a datasource at commit time, specifically when a single transaction requires data from multiple endpoints.</p> <p>If you use XA, the recovery log is stored in the tx_manager datasource, and its commit protocol must be optimistic. If tx_manager is aliased to another datasource (that is, one that is defined with the alias-For property), the commit protocol for that datasource must be optimistic. A last-resource optimization ensures full conformance with the XA specification. The commit protocol for all other datasources should be XA_2PC. Alternately, a transaction that accesses multiple datasources for which the commit protocols are optimistic is permitted.</p>	<p>[optimistic pessimistic XA_2PC]</p> <p>Choose only one of these protocols:</p> <ul style="list-style-type: none"> • Optimistic – enables connections to be committed without regard for other connections enlisted in the transaction, assuming that the transaction is not marked for rollback and will successfully commit on all resources. Note: if a transaction accesses multiple data sources with commit protocol of "optimistic", atomicity is not guaranteed. • Pessimistic – specifies that you do not expect any multi-resource transactions. An exception will be thrown (and transaction rolled back) if any attempt is made to use more than one "pessimistic" data source in the same transaction. • XA_2PC – specifies use of the XA two phase commit protocol. If you are using two phase commit, then the recovery log is stored in the "tx_manager" data source, and that data source (or the one it is aliased to) must have the commit protocol of "optimistic" or "pessimistic". All other data sources for which atomicity must be ensured should have the "XA_2PC" commit protocol.

Name	Description	Supported values
Datasource Class	<p>Sets the class that implements the JDBC datasource.</p> <p>Use this property (along with the driverClass property) only if you do not have a predefined database-type entry in Unwired Server for the kind of SQL database you are connecting to. For example, you must use this property for MySQL database connections.</p> <p>You can implement a datasource class to work with a distributed transaction environment. Because Unwired Server supports distributed transactions, some datasources may require that a datasource class be implemented for Unwired Server to interact with it.</p> <p>For two-phase transactions, use the xaDataSourceClass connection property instead.</p>	<p><com.mydata-source.jdbc.Driver></p>
Database Command Echo	<p>Echoes a database command to both the console window and the server log file.</p> <p>Use this property to immediately see and record the status or outcome of database commands.</p> <p>When you enable this property, Unwired Server echoes every SQL query to <code>ml.log</code>, which may help you debug your application.</p>	<p>[true false]</p> <p>Set a value of 1 to echo the database commands like <code>databaseStartCommand</code>, and <code>databaseStopCommand</code>.</p> <p>Otherwise, do not set this property, or use a value of 0 to disable the echo.</p>

Name	Description	Supported values
Database Create Command	Specifies the operating system command used to create the database for this datasource. If this command is defined and the file referenced by <code>\${databaseFile}</code> does not exist, the command is run to create the database when an application component attempts to obtain the first connection from the connection pool for this datasource.	<p><command></p> <p>Example: <UnwiredPlatform_InstallDir>\Servers\SQLAnywhere11\BIN32\dbinit -q <code>\${databaseFile}</code></p>
Database File	<p>Indicates the database file to load when connecting to a datasource.</p> <p>Use this property when the path to the database file differs from the one normally used by the database server.</p> <p>If the database you want to connect to is already running, use the <code>databaseName</code> connection parameter.</p>	<p><string></p> <p>Supply a complete path and file name. The database file you specify must be on the same host as the server.</p>

Name	Description	Supported values
Database Name	<p>Identifies a loaded database with which to establish a connection, when connecting to a datasource.</p> <p>Set a database name, so you can refer to the database by name in other property definitions for a datasource.</p> <p>If the database to connect to is not already running, use the database-File connection parameter so the database can be started.</p> <hr/> <p>Note: For Unwired Server, you typically do not need to use this property. Usually, when you start a database on a server, the database is assigned a name. The mechanism by which this occurs varies. An administrator can use the DBN option to set a unique name, or the server may use the base of the file name with the extension and path removed.</p>	<p>[DBN default]</p> <p>If you set this property to default, the name is obtained from the DBN option set by the database administrator.</p> <p>If no value is used, the database name is inherited from the database type.</p>
Database Start Command	Specifies the operating system command used to start the database for this datasource. If this command is defined and the database is not running, the command is run to start the database when the datasource is activated.	<p><command></p> <p>Example: <UnwiredPlatform_InstallDir>\Servers\SQLAnywhere11\BIN32\dbsrvXX.exe</p>
Database Stop Command	Specifies the operating system command used to stop the database for this datasource. If this property is defined and the database is running, this command executes during shutdown.	<p><command></p> <p>For a SQL Anywhere® database, where the user name and password are the defaults (dba and sql), enter:</p> <p><UnwiredPlatform_InstallDir>\Servers\SQLAnywhere11\BIN32\dbsrvXX.exe</p>

Name	Description	Supported values
Database Type	Specifies the database type.	<database type>
Database URL	<p>Sets the JDBC URL for connecting to the database if the datasource requires an Internet connection.</p> <p>Typically, the server attempts to construct the database URL from the various connection properties you specify (for example, portNumber, databaseName). However, because some drivers require a special or unique URL syntax, this property allows you to override the server defaults and instead provide explicit values for this URL.</p>	<p><JDBCurl></p> <p>The database URL is JDBC driver vendor-specific. For details, refer to the driver vendor's JDBC documentation.</p>
Driver Class	<p>Sets the name of the class that implements the JDBC driver.</p> <p>Use this property (along with the dataSourceClass property) only if you do not have a predefined database-type entry in Unwired Server for the kind of SQL database you are connecting to. For example, MySQL database connections require you to use this connection property.</p> <p>To create a connection to a database system, you must use the compatible JDBC driver classes. Sybase does not provide these classes; you must obtain them from the database manufacturer.</p>	<p><Class.forName("foo.bar.Driver")></p> <p>Replace <Class.forName("foo.bar.Driver")> with the name of your driver.</p>
Driver Debug	Enables debugging for the driver.	<p>[true false]</p> <p>Set to true to enable debugging, or false to disable.</p>
Driver Debug Settings	Configures debug settings for the driver debugger.	<p>[default <setting>]</p> <p>The default is STATIC:ALL.</p>

Name	Description	Supported values
Initial Pool Size	<p>Sets the initial number of connections in the pool for a datasource.</p> <p>In general, holding a connection causes a less dramatic performance impact than creating a new connection. Keep your pool size large enough for the number of concurrent requests you have; ideally, your connection pool size should ensure that you never run out of available connections.</p> <p>The initialPoolSize value is applied to the next time you start Unwired Server.</p>	<p><int></p> <p>Replace <int> with an integer to preallocate and open the specified number of connections at start-up. The default is 0.</p> <p>Sybase suggests that you start with 0, and create additional connections as necessary. The value you choose allows you to create additional connections before client synchronization requires the server to create them.</p>
Is Download Zipped	<p>Specifies whether the driver file downloaded from jdbcDriverDownloadURL is in .ZIP format.</p> <p>This property is ignored if the value of jdbcDriverDownloadURL connection is an empty string.</p>	<p>[True False]</p> <p>The default is false. The file is copied, but not zipped to <UnwiredPlatform-install>\lib\jdbc.</p> <p>Set isDownloadZipped to true to save the file to <UnwiredPlatform-install>\lib\jdbc and unzip the archived copy.</p>
JDBC Driver Download URL	<p>Specifies the URL from which you can download a database driver.</p> <p>Use this property with isDownloadZipped to put the driver in an archive file before the download starts.</p>	<p><URL></p> <p>Replace <URL> with the URL from which the driver can be downloaded.</p>

Name	Description	Supported values
Language	<p>For those interfaces that support localization, this property specifies the language to use when connecting to your target database. When you specify a value for this property, Unwired Server:</p> <ul style="list-style-type: none"> • Allocates a CS_LOCALE structure for this connection • Sets the CS_SYB_LANG value to the language you specify • Sets the Microsoft SQL Server CS_LOC_PROP connection property with the new locale information <p>Unwired Server can access Unicode data in an Adaptive Server® 12.5 or later, or in Unicode columns in Adaptive Server 12.5 or later. Unwired Server automatically converts between double-byte character set (DBCS) data and Unicode, provided that the Language and CodeSet parameters are set with DBCS values.</p>	<p><language></p> <p>Replace <language> with the language being used.</p>
Max Idle Time	<p>Specifies the number of seconds an idle connection remains in the pool before it is dropped.</p>	<p><int></p> <p>If the value is 0, idle connections remain in the pool until the server shuts down. The default is 60.</p>

Name	Description	Supported values
Max Pool Size	<p>Sets the maximum number of connections allocated to the pool for this datasource.</p> <p>Increase the <code>maxPoolSize</code> property value when you have a large user base. To determine whether a value is high enough, look for <code>ResourceMonitorTimeoutException</code> exceptions in <code><hostname>-server.log</code>. Continue increasing the value, until this exception no longer occurs.</p> <p>To further reduce the likelihood of deadlocks, configure a higher value for <code>maxWaitTime</code>.</p> <p>To control the range of the pool size, use this property with <code>minPoolSize</code>.</p>	<p><code><int></code></p> <p>A value of 0 sets no limit to the maximum connection pool size. The default is 10.</p>
Max Wait Time	Sets the maximum number of seconds to wait for a connection before the request is cancelled.	<p><code><int></code></p> <p>The default is 60.</p>
Max Statements	Specifies the maximum number of JDBC prepared statements that can be cached for each connection by the JDBC driver. The value of this property is specific to each JDBC driver.	<p><code><int></code></p> <p>A value of 0 (default) sets no limit to the maximum statements.</p>
Min Pool Size	Sets the minimum number of connections allocated to the pool for this datasource.	<p><code><int></code></p> <p>A value of 0 (default) sets no limit to the minimum connection pool size.</p>

Name	Description	Supported values
Network Protocol	<p>Sets the protocol used for network communication with the datasource.</p> <p>Use this property (along with the driverClass, and dataSourceClass properties) only if you do not have a predefined database-type entry in Unwired Server for the kind of SQL database you are connecting to. For example, you may be required to use this property for MySQL database connections.</p>	<p>The network protocol is JDBC driver vendor-specific. There are no predefined values.</p> <p>See the driver vendor's JDBC documentation.</p>
Password	Specifies the password for connecting to the database.	[default <password>]
Ping and Set Session Auth	Runs the ping and session-authorization commands in a single command batch; may improve performance. You can only enable the Ping and Set Session Auth property if you have enabled the Set Session Auth property so database work runs under the effective user ID of the client.	<p>[True False]</p> <p>Set to true to enable, or false to disable.</p>
Ping Connections	Pings connections before attempting to reuse them from the connection pool.	<p>[True False]</p> <p>Set to true to enable ping connections, or false to disable.</p>
Ping SQL	Specify the SQL statement to use when testing the database connection with ping.	<p>[default <statement>]</p> <p>Replace <statement> with the SQL statement identifier. The default is "select 1".</p>
Port Number	Sets the server port number where the database server listens for connection requests.	<p>[default <port>]</p> <p>Replace <port> with the TCP/IP port number to use (that is, 1 – 65535).</p> <p>If you set the value as default, the default protocol of the datasource is used.</p>

Name	Description	Supported values
PS Maximum Blob Length	Indicates the maximum number of bytes allowed when updating a BLOB datatype using Prepared-Statement.setBytes.	[default <int>] Replace <int> with the number of bytes allowed during an update. The default is 16384.
PS Maximum Clob Length	Indicates the maximum number of characters allowed when updating a CLOB datatype using Prepared-Statement.setString.	[default <int>] Replace <int> with the number of bytes allowed during an update. The default is 16384.
Role Name	Sets the database role that the user must have to log in to the database.	[default <name>] If you set this value to default, the default database role name of the data-source is used.
Server Name	Defines the host where the database server is running.	<name> Replace <name> with an appropriate name for the server.
Service Name	Defines the service name for the data-source. For SQL Anywhere servers, use this property to specify the database you are attaching to.	<name> Replace <name> with an appropriate name for the service.
Set Session Auth	Establishes an effective database identity that matches the current mobile application user. If you use this property, you must also use setSessionAuthSystemID to set the session ID. Alternately you can pingAndSetSessionAuth if you are using this property with pingConnection. The pingAndSetSessionAuth property runs the ping and session-authorization commands in a single command batch, which may improve performance.	[true false] Choose a value of 1 to use an ANSI SQL set session authorization command at the start of each database transaction. Set to 0 to use session-based authorizations.

Name	Description	Supported values
Set Session Auth System ID	If Set Session Authorization is enabled, specifies the database identity to use when the application server accesses the database from a transaction that runs with "system" identity.	<database identity> Replace <database identity> with the database identifier.
Start Wait	Sets the wait time (in seconds) before a connection problem is reported. If the start command completes successfully within this time period, no exceptions are reported in the server log. startWait time is used only with the databaseStartCommand property.	<int> Replace <int> with the number of seconds Unwired Server waits before reporting an error.
Truncate Nanoseconds	Sets a divisor/multiplier that is used to round the nanoseconds value in a java.sql.Timestamp to a granularity that the DBMS supports.	[default <int>] The default is 10 000 000.
Use Quoted Identifiers	Specifies whether or not SQL identifiers are quoted.	[True False] Set to true to enable use of quoted identifiers, or false to disable.
User	Identifies the user who is connecting to the database.	[default <user name>] Replace <user name> with the database user name.
XA Datasource Class	Specifies the class name or library name used to support two-phase commit transactions, and the name of the XA resource library.	<class name> Replace <class name> with the class or library name. <ul style="list-style-type: none"> • SQL Anywhere database: com.sybase.jdbc3.jdbc.SybXADataSource • Oracle database: oracle.jdbc.xa.client.OracleXADataSource

SAP Java Connector Properties

Configure SAP Java Connector (JCo) connection properties.

For a comprehensive list of SAP JCo properties you can use to create an instance of a client connection to a remote SAP system, see [http://help.sap.com/javadocs/NW04/current/jc/com/sap/mw/jco/JCO.html#createClient\(java.util.Properties\)](http://help.sap.com/javadocs/NW04/current/jc/com/sap/mw/jco/JCO.html#createClient(java.util.Properties)).

Table 8. General connection parameters

Name	Description	Supported values
Client Number	Specifies the SAP client.	Three-digit client number; preserve leading zeros if they appear in the number
Logon User	Specifies the login user ID.	User name for logging in to the SAP system If using X.509 certificate authentication, remove the JCo properties <code>jco.client.passwd</code> and <code>jco.client.user</code> defined for the SAP connection profile in Sybase Control Center (SCC).
Password	Specifies the login password.	Password for logging in to the SAP system
Language	Specifies a login language.	ISO two-character language code (for example, EN, DE, FR), or SAP-specific single-character language code. As a result, only the first two characters are ever used, even if a longer string is entered. The default is EN.
System Number	Indicates the SAP system number.	SAP system number
Host Name	Identifies the SAP application server.	Host name of a specific SAP application server
Message Server	Identifies the SAP message server.	Host name of the message server
Gateway Host	Identifies the SAP gateway host.	Host name of the SAP gateway Example: GWHOST=hs0311

Name	Description	Supported values
Gateway Service	Identifies the SAP gateway service.	Service name of the SAP gateway Example: GWSERV=sapgw53
R/3 Name	Specifies R/3 name.	Name of the SAP system
Server Group	Identifies the group of SAP application servers.	Group name of the application servers
External Server Program	Identifies the program ID of the external server program.	Path and name of the external RFC server program, or program ID of a registered RFC server program Example: TPNAME= /sap/ srfcserv
External Server Program Host	Identifies the host of the external server program. This information determines whether the RFC client connects to an RFC server started by the SAP gateway or to an already registered RFC server. Note: If the gateway host and external server program host are different, make sure that the SAP gateway has access to start the server program through a remote shell.	Host name of the external RFC server program Example: TPHOST=hs0311
Remote Host Type	Identifies the type of remote host.	2: R/2 3: R/3 E: external
RFC Trace	Specifies whether or not to enable RFC trace.	0: disable 1: enable
Initial Codepage	Identifies the initial code page in SAP notation. A code page is used whenever character data is processed on the application server, appears on the front end, or is rendered by a printer.	Four-digit SAP code page number

Name	Description	Supported values
Enable ABAP Debugging	<p>Enables or disables ABAP debugging. If enabled, the connection is opened in debug mode and the invoked function module can be stepped through in the debugger.</p> <p>For debugging, an SAP graphical user interface (SAPGUI) must be installed on the same machine the client program is running on. This can be either a normal Windows SAPGUI or a Java GUI on Linux/UNIX systems.</p>	<p>0: no debugging</p> <p>1: attach a visible SAPGUI and break at the first ABAP statement of the invoked function module</p>
Remote GUI	<p>Specifies whether a remote SAP graphical user interface (SAPGUI) should be attached to the connection. Some older BAPIs need an SAPGUI because they try to send screen output to the client while executing.</p>	<p>0: no SAPGUI</p> <p>1: attach an "invisible" SAPGUI, which receives and ignores the screen output</p> <p>2: attach a visible SAPGUI</p> <p>For values other than 0 a SAPGUI needs to be installed on the machine, where the client program is running. This can be either a Windows SAPGUI or a Java GUI on Linux/Unix systems.</p>
Get SSO Ticket	<p>Generates an SSO2 ticket for the user after login to allow single sign-on. If RfcOpenConnection() succeeds, you can retrieve the ticket with RfcGetPartnerSSOTicket() and use it for additional logins to systems supporting the same user base.</p>	<p>0: do not generate SSO2 ticket</p> <p>1: generate SSO2 ticket</p>
Use Cookie Version 2	<p>Indicates whether or not to use the specified SAP Cookie Version 2 (SSO2) as the login ticket instead of user ID and password.</p>	<p>User: \$MYSAPSSO2\$</p> <p>Password: Base64-encoded ticket</p> <p>Login with single sign-on is based on secure network connection (SNC) encryption and can only be used in combination with an SNC.</p>

Name	Description	Supported values
Use X509	Indicates whether or not to use the specified X509 certificate as the login certificate instead of user ID and password.	User: \$X509CERT\$ Password: Base64-encoded ticket Login with X509 is based on secure network connection (SNC) encryption and can only be used in combination with an SNC.
Logon Check	Enables or disables login check at open time.	0: disable 1: enable If you set this to 0, RfcOpenConnection() opens a network connection, but does not perform the login procedure. Therefore, no user session is created inside the back-end system. This parameter is intended only for executing the function module RFC_PING.
Additional GUI Data	Provides additional data for graphical user interface (GUI) to specify the SAProuter connection data for the SAPGUI when it is used with RFC.	/H/ <i>router string</i> : the entire router string for the SAPGUI /P/ <i>password</i> : specify this value if the password for the SAPGUI connection is not the same as the password for the RFC connection.
GUI Redirect Host	Identifies which host to redirect the remote graphical user interface to.	Host name
GUI Redirect Service	Identifies which service to redirect the remote graphical user interface to.	Name of the service
Remote GUI Start Program	Indicates the program ID of the server that starts the remote graphical user interface.	Program ID of the server
SNC Mode	Enables or disables secure network connection mode.	0: off 1: on
SNC Partner	Identifies the secure network connection partner.	Secure network connection name of the application server (for example, p:CN=R3, O=XYZ-INC, C=EN)

Name	Description	Supported values
SNC Level	Specifies the secure network connection security level.	1: digital signature 2: digital signature and encryption 3: digital signature, encryption, and user authentication 8: default value defined by backend system 9: maximum value that the current security product supports
SNC Name	Indicates the secure network connection name. This property overrides the default secure network connection partner.	Token or identifier representing the external RFC program
SNC Service Lib Path	Identifies the path to the SAP cryptographic library that provides secure network connection service.	Full path and name of third-party security library. You must download and install the library from the SAP Service Marketplace.
R/2 Destination	Identifies a configured R/2 system defined in the sideinfo configuration.	
Logon ID	Defines the string for SAPLOGON on 32-bit Windows.	String key to read parameters from the saplogon.ini file created by the SAPLogon GUI program on Windows
External Authentication Data	Provides data for external authentication (PAS). This is an old login mechanism similar to SSO; Sybase recommends that you do not use this approach.	
External Authentication	Specifies type of external authentication (PAS). See External Authentication Data property.	

SAP DOE-C Properties

Configure SAP Data Orchestration Engine Connector (DOE-C) properties. This type of connection is available in the list of connection templates only when you deploy a DOE-C package. No template exists for these types of connections.

Note: If you change the username or password property of a DOE-C connection, you must reopen the same dialog and click **Test Connection** after saving. Otherwise the error state

of this DOE-C package is not set properly, and an error message is displayed. This will not work if you click **Test Connection** before saving the properties.

Name	Description	Supported values
Username	<p>Specifies the SAP user account ID. The SAP user account is used during interaction between the connected SAP system and client for certain administrative activities, such as sending acknowledgment messages during day-to-day operations or "unsubscribe" messages if a subscription for this connection is removed.</p> <p>This account is not used for messages containing business data; those types of messages are always sent within the context of a session authenticated with credentials provided by the mobile client.</p> <p>The technical user name and password or certificateAlias must be set to perform actions on subscriptions. The certificateAlias is mutually exclusive with and overrides the technical user name and password fields if set. The technical user name and password fields can be empty, but only if certificateAlias is set.</p>	Valid SAP login name for the DOE host system.
Password	Specifies the password for the SAP user account.	Valid password.
DOE SOAP Timeout	Specifies a timeout window during which unresponsive DOE requests are aborted.	Positive value (in seconds). The default is 420 (7 minutes).

Name	Description	Supported values
DOE Extract Window	Specifies the number of messages allowed in the DOE extract window.	<p>Positive value (in messages). The default is 50.</p> <p>When the number of messages in the DOE extract window reaches 50% of this value, DOE-C sends a <code>Status-ReqFromClient</code> message, to advise the SAP DOE system of the client's messaging status and acknowledge the server's state. The default value is 50.</p>
Packet Drop Size	<p>Specifies the size, in bytes, of the largest JavaScript Object Notation (JSON) message that the DOE connector processes on behalf of a JSON client.</p> <p>The packet drop threshold size should be carefully chosen, so that it is larger than the largest message sent from the DOE to the client, but smaller than the maximum message size which may be processed by the client.</p>	<p>Positive value (in bytes). The default is 1048576 bytes (1MB). Do not set lower than 4096 bytes; there is no maximum limitation.</p>
Service Address	Specifies the DOE URL.	<p>Valid DOE URL.</p> <p>If you are using DOE-C with SSO:</p> <ul style="list-style-type: none"> • Modify the port from the standard <code>http://host:8000</code> to <code>https://host:8001/</code>. • Add the certificate being used as the technical user and DOE-C endpoint security profile certificate to the SAP DOE system's SSL Server certificate list by using the <code>STRUST</code> transaction. See your SAP documentation for details.
Listener URL	Specifies the DOE-C server listener URL.	Valid DOE-C listener URL, for example <code>http://<sup_host-name>:8000/doi/publish</code> .

Name	Description	Supported values
SAP Technical User Certificate Alias	<p>Sets the alias for the Unwired Platform keystore entry that contains the X.509 certificate for Unwired Server's SSL peer identity.</p> <p>If you do not set a value, mutual authentication for SSL is not used when connecting to the Web service.</p> <p>If you are using DOE-C with SSO use the "SAP Technical User Certificate Alias" only for configurations which require the technical user to identify itself using an X.509 certificate; it specifies the Certificate Alias to be used as the technical user. This overrides the "Username" and "Password" settings normally used.</p>	Valid certificate alias.
Login Required	<p>Indicates whether authentication credentials are required to login. The default value is true.</p> <p>For upgraded packages, "login-required=false" gets converted to "login-required=true" and a No-Auth security configuration "DOECNoAuth" is assigned to the upgraded package.</p>	A read-only property with a value of true.

Web Services Properties

Configure connection properties for the Simple Object Access Protocol (SOAP) and Representational State Transfer (REST) architectures.

Name	Description	Supported Values
Password	Specifies the password for HTTP basic authentication, if applicable.	Password
Address	Specifies a different URL than the port address indicated in the WSDL document at design time.	HTTP URL address of the Web service
User	Specifies the user name for HTTP basic authentication, if applicable.	User name

Name	Description	Supported Values
Certificate Alias	Sets the alias for the Unwired Platform keystore entry that contains the X.509 certificate for Unwired Server's SSL peer identity. If you do not set a value, mutual authentication for SSL is not used when connecting to the Web service.	Use the alias of a certificate stored in the Unwired Server certificate keystore.

Proxy Properties

(Applies only to Online Data Proxy) Proxy properties identify the application endpoint and the pool size.

Name	Description	Supported values
User	Not currently used	
Certificate Alias	Not currently used	
Address	Corresponds to the Application end-point provided at the time of registering an application.	Must be a valid application end-point.
Pool Size	Determines the maximum number of connections allocated to the pool for this datasource.	The default value set for the pool size is 25.
Password	Not currently used	

Note: When the application end-point for a registered application is modified under the **Applications** node, you have to manually update the **Address** in the proxy properties of the connection pool.

Configuring Domain Security

Configure security for an individual domain to meet the customer's security requirements.

Prerequisites

Before mapping and assigning administrator roles, ensure that you have set the Unwired Platform administration and user roles and passwords required for Sybase Control Center administrator login. See *Enabling Authentication and RBAC for Administrator Logins*.

Task

Perform steps to appropriately configure domain security settings.

Choosing a Security Configuration

Select a security configuration that designates authentication, authorization, attribution, and audit security providers for the packages in the domain. You can assign as many security configurations as needed to a domain.

Only super administrators have privileges to create security configurations. Domain administrators can view a security configuration only after a super administrator has assigned it to the domain.

1. In the left navigation pane, navigate to **Cluster > Domains > <DomainName> > Security**.
2. In the right administration pane, select the **Security Configurations** tab and click **Assign**.
The **Assign Security Configurations** dialog appears.
3. Select one or more security configurations to assign to the domain by checking the box adjacent to the configuration name.
4. Click **OK**.
A message appears above the right administration pane menu indicating the success or failure of the assignment. If successful, the new security configuration appears in the list of security configurations.
5. To remove a security configuration, check the box adjacent to the configuration name and click **Unassign**. If a security configuration is mapped to one or more MBO packages, it can not be removed.

Assigning Domain Administrators to a Domain

Assign domain administration privileges to a domain administrator. You must be a platform administrator to assign and unassign domain administrators.

Prerequisites

Ensure the user is already registered as a domain administrator in the Domain Administrators tab.

Task

1. In the left navigation pane, expand the **Domains** folder, and select the domain for which to assign domain administration privileges.
2. Select the domain-level **Security** folder.
3. In the right administration pane, select the **Domain Administrators** tab, and click **Assign**.
4. Select one or more administrator users to assign to the domain by checking the box adjacent to the user name.

5. Click **OK**.

A message appears above the right administration pane menu indicating the success or failure of the assignment. If successful, the new domain administrator appears in the list of users.

Mapping Roles

Configure role mapping to authorize client requests to access MBOs and operations. For each security configuration, platform and domain administrators can manage logical role mappings at the package level or at a domain level. Use the corresponding domain or package node in the left navigation pane to configure role mappings accordingly.

Set an appropriate mapping state for each logical role. The state you choose allows you to disable logical roles, allow logical roles to be automatically mapped, or manually define which logical roles are mapped to one or more physical roles. The states of AUTO or NONE require the least administration.

If a developer has defined a logical role, mapping is not required; the logical role is matched to the physical role of the same name and is therefore automatically mapped.

Note: Changes to domain-level role mapping are applied to all domains that share the same security configuration. Likewise, changes to package-level role mapping apply to all instances of the affected package that use the same security configuration, even if the package is deployed in multiple domains.

Setting the Mapping State

Map roles for a package by setting the mapping state. Mapping behavior is determined by the state that exists for the logical role. You can select AUTO or NONE; a third state, MAPPED, is set automatically after you manually map a physical role to the selected logical role.

You can set the mapping state either when managing roles, or earlier, during package deployment. If your logical roles for a package do not automatically match the role names registered in the back-end security system, map corresponding logical and physical names to ensure that users can be authorized correctly.

1. For package-specific role mapping, select and deploy an available package. Follow the wizard prompts until you reach the Configure Role Mapping page for the target package.
2. Change the mapping for a logical role, if required:
 - To change the state to either NONE or AUTO, click the list adjacent to the logical role and click the appropriate option.
 - To change the role mapping itself, click the drop-down list adjacent to the logical role and choose **Map Role**. This command displays the Role Mappings dialog that allows you to manually set the physical role mappings. The Role Mappings dialog displays the name of the logical role you are mapping in the text area of the dialog. Once saved, the state automatically changes to MAPPED.
3. Click **Next**.

The Server Connection page appears.

Deployment-time role mapping is done at the package level. Once the package is deployed, you can change the role mapping by going to the Role Mapping tab for the desired package. You can also set the role mapping for each security configuration at the domain level. This allows the role mapping to be shared across packages for the common logical roles. Changing role mapping at the domain level will result in role mapping changes in other domains where the same security configuration is referenced.

Mapping a Physical Role Manually

Use the Role Mappings dialog to manually map required physical roles for a logical role when physical and logical role names do not match. If names do not match, the AUTO mapping state does not work.

Prerequisites

Unwired Platform cannot query all supported enterprise security servers directly; for successful authentication, you must know the physical roles your back-end systems require.

Task

You can map a logical role to one or more physical roles. You can also map multiple logical roles to the same physical role. If a role does not exist, you can also add or delete names as needed.

1. Review the list of existing physical role names that you can map to the logical role you have selected.
2. If a role that you require does not appear, enter the **Role name** and click the + button. The role name appears in the **Available roles** list with an asterisk (*). This asterisk indicates that an available role was added by an administrator, not a developer.
3. To remove a role you no longer require from the **Available roles** list, select the name and click the **x** button adjacent to the **Role name** field. The role is removed and can no longer be mapped to a logical role.
4. To map a logical role that appears in the text area of the Role Mappings dialog to a physical role:
 - a) Select one or more **Available roles**.
 - b) Click **Add**.
5. To unmap a role:
 - a) Select one or more **Mapped roles**.
 - b) Click **Remove**.
The roles are returned to the **Available roles** list.
6. Click **OK** to save these changes.

Administer

Once a logical role has been manually mapped, the mapping state changes to MAPPED. The roles you have mapped appear in the active Physical Roles cell for either a package-specific or server-wide role mappings table.

Mapping State Reference

The mapping state determines the authorization behavior for a logical name instance.

State	Description
AUTO	Map the logical role to a physical role of the same name. The logical role and the physical role must match, otherwise, authorization fails.
NONE	Disable the logical role, which means that the logical role is not authorized. This mapping state prohibits anyone from accessing the resource (MBO or Operation). Carefully consider potential consequences before using this option.
MAPPED	A state that is applied after you have actively mapped the logical role to one or more physical roles. Click the cell adjacent to the logical role name and scroll to the bottom of the list to see the list of mapped physical roles.

Security Configurations

Sybase Unwired Platform does not provide proprietary security systems for storing and maintaining users and access control rules, but delegates these functions to the enterprise's existing security solutions.

A security configuration determines the scope of user identity, performs authentication and authorization checks, and can be assigned multiple levels (domain or package). Applications inherit a security configuration when the administrator assigns the application to a domain via a connection template.

Users can be authenticated differently, depending on which security configuration is used. For example, a user identified as "John" may be authenticated different ways, depending on the named security configuration protecting the resource he is accessing: it could be an MBO package, a DCN request, use of Sybase Control Center .

Security configurations aggregate various security mechanisms for protecting Unwired Platform resources under a specific name, which administrators can then assign. Each security configuration consists of:

- A set of configured security providers. Security provider plug-ins for many common security solutions are included with the Sybase Unwired Platform.
- Role mappings (which are set at the domain and package level) that map logical roles to back end physical roles.

A user entry must be stored in the security repository used by the configured security provider to access any resources (that is, either a Sybase Control Center administration feature or an

application package that accesses data sets from a back-end data source). When a user attempts to access a particular resource, Unwired Server tries to authenticate and authorize the user, by checking the security repository for:

- Security access policies on the requested resource
- Role memberships

Creating a Security Configuration

Create and name a set of security providers and physical security roles to protect Unwired Platform resources.

Only platform administrators can create security configurations. Domain administrators can only view after the platform administrator creates and assigns them to a domain.

1. In the left navigation pane of Sybase Control Center, select **Security**.
2. In the right administration pane, click **New**.
3. Enter a name for the security configuration and click **OK**.
4. In the left navigation pane, under **Security**, select the new security configuration.
5. In the right administration pane, select the Settings tab.

The **Authentication cache timeout** determines how long authentication results should be cached before a user is required to reauthenticate. For details, see *Authentication Cache Timeouts* in *Security*. To configure this value:

- a) Set the cache timeout value in seconds. The default is 3600. To force re-authentication, change this value to 0.

The **Maximum allowed authentication failure** determines the maximum number of login attempts after which the user is locked. To configure this value:

- a) Set the maximum count for authentication failure.

The **Authentication lock duration** determines how long the user is locked after the maximum login attempts is reached . To configure this value:

- a) Set the authentication lock duration in seconds.

6. Click **Save**.
7. Select the tab corresponding to the type of security provider you want to configure: Authentication, Authorization, or Audit.
8. To edit the properties of a preexisting security provider in the configuration:
 - a) Select the provider, and click **Properties**.
 - b) Configure the properties associated with the provider by setting values according to your security requirements. Add properties as required. For more information about configuring security provider properties, see the individual reference topics for each provider.
 - c) Click **Save**.
9. To add a new security provider to the configuration:

- a) Click **New**.
- b) Select the provider you want to add.
- c) Configure the properties associated with the provider by setting values according to your security requirements. Add properties as required. For more information about configuring security provider properties, see the individual reference topics for each provider.
- d) Click **OK**.

The configuration is saved locally, but not yet committed to the server.

10. Select the **General** tab, and click **Validate** to confirm that Unwired Server accepts the new security configuration.

A message indicating the success of the validation appears above the menu bar.

11. Click **Apply** to save changes to the security configuration, and apply them across Unwired Server.

A message indicating the success of the application appears above the menu bar.

Security Providers

Different security providers give Unwired Server security features that include authentication, and authorization capabilities.

Configure security providers for Unwired Server by logging in to the server in Sybase Control Center and clicking **Security > Configuration**. Configuring these providers writes changes to the Unwired Server configuration properties file.

For third-party providers, save related JAR files or DLLs in the
<UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers
\UnwiredServer\lib\ext folder.

- Authentication modules – verify the identity of a user accessing a network with the mobile application, typically via a login form or some other login or validation mechanism. Authentication in Unwired Server is distinct from authorization. You must have at least one authentication module configured in a production deployment of Unwired Server. You can stack multiple providers so users are authenticated in a particular sequence.
- Authorization modules – check the access privileges for an authenticated identity. Sybase recommends that you have at least one authorization module configured in a production deployment of Unwired Server.
- Auditing modules – report all audit events to allow you to evaluate the security system implementation for Unwired Server. Auditing provides you a record of all the security decisions that have been made. Each successful authentication creates a session key that shows up in subsequent security checks for that user. Unsuccessful authentications are also logged. Each authorization records what roles were checked, or what resource was accessed. Audit filters determine what events get recorded, the audit format determines what the audit records look like, and the audit destination specifies where audit records are sent. Use the audit trail to identify who did what and when, with respect to objects secured by your providers. Auditing modules are optional.

In most cases, each security module requires a unique set of configuration properties. However, there are some cases when modules require a common set of properties, and these properties are configured once for each module on a tab created for that purpose.

You can configure different security providers for administrator authentication and device user authentication. For more information on configuring security providers depending on the type of user, see either *Enabling Authentication and RBAC for User Logins* or *Enabling Authentication and RBAC for Administrator Logins* in the *Security* guide.

Stacking Providers and Combining Authentication Results

(Not applicable to Online Data Proxy) Optionally, implement multiple login modules to provide a security solution that meets complex security requirements. Sybase recommends provider stacking as a means of eliciting more precise results, especially for production environment that require different authentications schemes for administrators, DCN, SSO, and so on.

Stacking is implemented with a controlFlag attribute that controls overall behavior when you enable multiple providers. Set the controlFlag on a specific provider to refine how results are processed.

For example, say your administrative users (supAdmin in a default installation) are not also users in an EIS system like SAP. However, if they are authenticated with just the default security configuration, they cannot also authenticate to the SAPSSOTokenLoginModule used for SSO2Token retrieval. In this case, you would stack a second login modules with a controlFlag=sufficient login module for your administrative users.

Or, in a custom security configuration (recommended), you may also find that you are using a technical user for DCN who is also not an SAP user. This technical user does not need SSO because they will not need to access data. However, the technical user still needs to be authenticated by Unwired Server. In this case, you can also stack another login module so this DCN user can login.

1. Use Sybase Control Center to create a security configuration and add multiple providers as required for authentication.
2. Order multiple providers by selecting a login module and using the up or down arrows at to place the provider correctly in the list.

The order of the list determines the order in which authentication results are evaluated.

3. For each provider:
 - a) Select the provider name.
 - b) Click **Properties**.
 - c) Configure the controlFlag property with one of the available values: required, requisite, sufficient, optional.

See *controlFlag Attribute Values* for descriptions of each available value.

- d) Configure any other common security properties as required.

Administer

4. Click **Save**.
5. Select the **General** tab, and click **Apply**.

For example, say you have sorted these login modules in this order and used these controlFlag values:

- LDAP (required)
- NT Login (sufficient)
- SSO Token (requisite)
- Certificate (optional)

The results are processed as indicated in this table:

Pro-vider	Authentication Status							
LDAP	pass	pass	pass	pass	fail	fail	fail	fail
NT Log-in	pass	fail	fail	fail	pass	fail	fail	fail
SSO To-ken	*	pass	pass	fail	*	pass	pass	fail
Certifi-cate	*	pass	fail	*	*	pass	fail	*
Overall result	pass	pass	pass	fail	fail	fail	fail	fail

Stacking LoginModules in SSO Configurations

(Not applicable to Online Data Proxy) Use LoginModule stacking to enable role-based authorization for MBOs and data change notifications (DCNs).

controlFlag Attribute Values

(Not applicable to Online Data Proxy) The Sybase implementation uses the same controlFlag values and definitions as those defined in the JAAS specification.

If you stack multiple providers, you must set the controlFlag attribute for each enabled provider.

Control Flag Value	Description
(Default) required	The LoginModule is required. Authentication proceeds down the LoginModule list.

Control Flag Value	Description
requisite	<p>The LoginModule is required. Subsequent behavior depends on the authentication result:</p> <ul style="list-style-type: none"> • If authentication succeeds, authentication continues down the LoginModule list. • If authentication fails, control returns immediately to the application (authentication does not proceed down the LoginModule list).
sufficient	<p>The LoginModule is not required. Subsequent behavior depends on the authentication result:</p> <ul style="list-style-type: none"> • If authentication succeeds, control returns immediately to the application (authentication does not proceed down the LoginModule list). • If authentication fails, authentication continues down the LoginModule list.
optional	<p>The LoginModule is not required. Irrespective of success or failure, authentication proceeds down the LoginModule list.</p>

Example

Providers are listed in this order and with these controlFlag:

1. CertificateAuthenticationLoginModule (sufficient)
2. LDAP (optional)
3. NativeOS (sufficient)

A client doing certificate authentication (for example, X.509 SSO to SAP) can authenticate immediately. Subsequent modules are not called, because they are not required. If there are regular username/password credentials, they go to LDAP, which may authenticate them, and set them up with roles from the LDAP groups they belong to. Then NativeOS is invoked, and if that also succeeds, Unwired Platform picks up roles based on the Windows groups they are in.

LDAP Login and Authorization Modules

LDAP login and authorization modules can sometimes share a common configuration. However, authorizers do not inherit configuration from login modules you configure. Configurations must be explicit.

In the case where both LDAPLoginModule and LDAPAuthorizer are configured:

- Matching configuration – LDAPAuthorizer simply skips the role retrieval.
- Differing configuration – LDAPAuthorizer proceeds with the role retrieval from the configured backend, and performs the authorization checks using the complete list of roles (from both the login module and itself). Even in the case where multiple

LDAPLoginModules are configured, only one LDAPAuthorizer is required as it compares its configuration with the configuration used for the successful authentication of the user.

Reordering Configured Providers

List stacked security providers for a security configuration to identify them as primary or auxiliary providers. Authentication, authorization, or attribution by provider take place in the order in which the providers are listed.

1. In the left navigation pane, expand the **Security** folder.
2. Select the security configuration you want to administer.
3. In the right administration pane, select the tab corresponding to the type of security provider you want to configure: Authentication, Authorization, Attribution, or Audit.
4. Select a provider from the list, then use the up and down arrows to the right of the table to achieve the desired placement.
5. Click **Save**.
6. Select the **General** tab, and click **Apply**.

A notification message appears if a server restart is required for changes to take effect.

Security Provider Configuration Properties

Security providers implement different properties, depending on whether or not they support authentication, authorization, audit, or attribution.

Platform administrators can configure application security properties in the Sybase Control Center for Unwired Platform console. These properties are then transcribed to an XML file in the <UnwiredPlatform_InstallDir>\Servers\UnwiredServer\Repository\CSI\ directory. A new section is created for each provider you add.

NoSec Configuration Properties

A NoSec provider offers pass-through security for Unwired Server. In development environments, you can apply a NoSec security provider for authentication, authorization, and attribution modules. However, never use this provider in production environments — either for administration, or device user authentication.

- The NoSecLoginModule class provides open authentication services
- The NoSecAuthorizer class provides authorization services
- The NoSecAttributer provides attribution services

However, you need to configure only authentication properties for a NoSec provider.

Table 9. Authentication properties

Property	Default Value	Description
useUsernameAsIdentity	true	If this option is set to true, the user name supplied in the callback is set as the name of the principal added to the subject.
identity	nosec_identity	The value of this configuration option is used as the identity of the user if either of these conditions is met: <ul style="list-style-type: none"> • No credentials were supplied. • The useUsernameAsIdentity option is set to false.
useFirstPass	false	If set to true, the login module attempts to retrieve only the user name and password from the shared context. It never calls the callback handler.
tryFirstPass	false	If set to true, the login module first attempts to retrieve the user name and password from the shared context before attempting the callback handler.
clearPass	false	If set to true, the login module clears the user name and password in the shared context when calling either commit or abort.
storePass	false	If set to true, the login module stores the user name and password in the shared context after successfully authenticating.

No Security Provider

A NoSec provider offers pass-through security for Unwired Server, and is intended for use in development environments or for deployments that require no security control. Do not use this

provider in production environments— either for administration, or device user authentication.

If you use NoSec providers, all login attempts succeed, no matter what values are used for the user name and password. Additionally, all role and control checks based on attributes also succeed.

Sybase provides these classes to implement the NoSec provider:

- **NoSecLoginModule** – provides pass-through authentication services.
- **NoSecAttributer** – provides pass-through attribution services.
- **NoSecAuthorizer** – provides pass-through authorization services.

Disabling Security with the NoSec Provider

Disable security using the NoSec provider.

1. In the left navigation pane, expand the **Security** folder.
2. In the right administration pane, click **New**.
3. Enter a name for the No Security configuration and click **OK**.
4. In the left navigation pane, under **Security**, select the new security configuration.
5. In the right administration pane, select **com.sybase.security.core.NoSec<ModuleType>** as the security provider for authentication and attribution. Leave the security provider for authorization and audit empty.
 - a) Select the tab corresponding to the appropriate security type.
 - b) Click **New**.
 - c) From the menu, select **com.sybase.security.core.NoSec<ModuleType>**.
 - d) Configure the properties associated with the provider by setting values according to your requirements. Add properties as required. For more information about configuring NoSec security provider properties, see the individual reference topic.
 - e) Click **OK**.
6. Select the **General** tab, and click **Validate** to confirm that Unwired Server accepts the new security configuration.

A message indicating the success of the validation appears above the menu bar.
7. Click **Apply** to save changes to the security configuration, and apply them across Unwired Server.

A message indicating the success of the application appears above the menu bar.

Next

Assign the newly created security configuration to the domain where the package that requires no security is deployed.

LDAP Configuration Properties

(Not applicable to Online Data Proxy) Use these properties to configure the LDAP provider used to authenticate SCC administration logins or to configure the LDAP provider used to

authenticate device application logins. If you are creating a provider for device application logins, then Unwired Platform administrators use Sybase Control Center to write these properties to the `<UnwiredPlatform_InstallDir>\Servers\UnwiredServer\Repository\CSI\default.xml` file.

Unwired Server implements a Java LDAP provider through a common security interface used by other Sybase products like Sybase Control Center.

The Java LDAP provider consists of three provider modules, each of which is in the `com.sybase.security.ldap` Java package. This is why the syntax used between Sybase Control Center provider and Unwired Server varies.

- **LDAPLoginModule**— provides authentication services. Through appropriate configuration, you can enable certificate authentication in **LDAPLoginModule**.
- (Optional)**LDAPAuthorizer** or **RoleCheckAuthorizer** — provide authorization services for **LDAPLoginModule**. **LDAPLoginModule** works with either authorizer. In most production deployments, you must always configure your own authorizer. However, if you are authenticating against a service other than LDAP, but want to perform authorization against LDAP, you can use the **LDAPAuthorizer**.

The **RoleCheckAuthorizer** is used with every security configuration but does not appear in Sybase Control Center.

Use **LDAPAuthorizer** only when **LDAPLoginModule** is not used to perform authentication, but roles are still required to perform authorization checks against the LDAP data store. If you use **LDAPAuthorizer**, always explicitly configure properties; for it cannot share the configuration options specified for the **LDAPLoginModule**.

Use this table to help you configure properties for one or more of the supported LDAP providers. When configuring modules or general server properties in Sybase Control Center, note that properties and values can vary, depending on which module or server type you configure.

Property	Default Value	Description
ServerType	None	<p>Optional. The type of LDAP server you are connecting to:</p> <ul style="list-style-type: none"> • sunone5 -- SunOne 5.x OR iPlanet 5.x • msad2k -- Microsoft ActiveDirectory, Windows 2000 • nsds4 -- Netscape Directory Server 4.x • openldap -- OpenLDAP Directory Server 2.x <p>The value you choose establishes default values for these other authentication properties:</p> <ul style="list-style-type: none"> • RoleFilter • UserRoleMembership • RoleMemberAttributes • AuthenticationFilter • DigestMD5Authentication • UseUserAccountControl
ProviderURL	ldap://localhost:389	<p>The URL used to connect to the LDAP server. Without this URL configured, Unwired Server cannot contact your server. Use the default value if the server is:</p> <ul style="list-style-type: none"> • Located on the same machine as your product that is enabled with the common security infrastructure. • Configured to use the default port (389). <p>Otherwise, use this syntax for setting the value:</p> <p>ldap://<hostname>:<port></p>

Property	Default Value	Description
DefaultSearchBase	None	<p>The LDAP search base that is used if no other search base is specified for authentication, roles, attribution and self registration:</p> <ol style="list-style-type: none"> 1. <code>dc=<domainname>,dc=<tld></code> For example, a machine in sybase.com domain would have a search base of <code>dc=sybase,dc=com</code>. 2. <code>o=<company name>,c=<country code></code> For example, this might be <code>o=Sybase,c=us</code> for a machine within the Sybase organization.
SecurityProtocol	None	<p>The protocol to be used when connecting to the LDAP server.</p> <p>To use an encrypted protocol, use "ssl" instead "ldaps" in the url.</p> <hr/> <p>Note: ActiveDirectory requires the SSL protocol when setting the value for the password attribute. This occurs when creating a user or updating the password of an existing user.</p> <hr/>
AuthenticationMethod	simple	<p>The authentication method to use for all authentication requests into LDAP. Legal values are generally the same as those of the <code>java.naming.security.authentication</code> JNDI property. Choose one of:</p> <ul style="list-style-type: none"> • simple — For clear-text password authentication. • DIGEST-MD5 — For more secure hashed password authentication. This method requires that the server use plain text password storage and only works with JRE 1.4 or later.

Property	Default Value	Description
AuthenticationFilter	<p>For most LDAP servers: (&(uid={uid}) (object- class=person))</p> <p>or</p> <p>For Active Directory email lookups: (&(userPrinci- palName={uid}) (object- class=user)) [ActiveDirec- tory]</p> <p>For Active Directory Windows username lookups: (&(SAMAc- count- Name={uid}) (object- class=user))</p>	<p>The filter to use when looking up the user.</p> <p>When performing a username based lookup, this filter is used to determine the LDAP entry that matches the supplied username.</p> <p>The string "{uid}" in the filter is replaced with the supplied username.</p>
AuthenticationScope	onelevel	<p>The authentication search scope. The supported values for this are:</p> <ul style="list-style-type: none"> • onellevel • subtree <p>If you do not specify a value or if you specify an invalid value, the default value is used.</p>
AuthenticationSearchBase	none	<p>The search base used to authenticate users. If this value is not specified, the LDAP DefaultSearch-Base is used.</p>

Property	Default Value	Description
BindDN	none	<p>The user DN to bind against when building the initial LDAP connection.</p> <p>In many cases, this user may need read permissions on all user records. If you do not set a value, anonymous binding is used. Anonymous binding works on most servers without additional configuration.</p> <p>However, the LDAP attributer may also use this DN to create the users in the LDAP server. When the self-registration feature is used, this user may also need the requisite permissions to create a user record. This behavior can occur if you do not set <code>useUserCredentialsToBind</code> to <code>true</code>. In this case, the LDAP attributer uses this DN to update the user attributes.</p>
BindPassword	none	<p>BindPassword is the password for BindDN, which is used to authenticate any user. BindDN and BindPassword are used to separate the LDAP connection into units.</p> <p>The <code>AuthenticationMethod</code> property determines the bind method used for this initial connection.</p> <p>If you use an encrypted the password using the CSI encryption utility, append <code>.e</code> to the property name. For example:</p> <pre>CSI.loginModule.7.options. BindPassword.e=1-AAAAEgQQOLL+LpX J08f09T4SrQYRC9lRT1w5ePfdczQTDs P8iACk9mDAbm3F3p5a1wXWKK8+NdJuk nc7w2nw5aGJlyG3xQ==</pre>
RoleSearchBase	none	<p>The search base used to retrieve lists of roles. If this value is not specified, the LDAP <code>DefaultSearchBase</code> is used.</p>

Property	Default Value	Description
RoleFilter	<p>For SunONE/iPlanet: (&(object-class=ldapsu-bentry) (objectclass=nsroledefinition))</p> <p>For Netscape Directory Server: (object-class=groupof-names) (object-class=groupofuniquenames))</p> <p>For ActiveDirectory: (object-class=groupof-names) (object-class=group))</p>	<p>The role search filter. This filter should, when combined with the role search base and role scope, return a complete list of roles within the LDAP server. There are several default values depending on the chosen server type. If the server type is not chosen or this property is not initialized, no roles are available.</p>
RoleMemberAttributes	<p>For Netscape Directory Server: member,unique-member</p>	<p>The role's member attributes defines a comma-delimited list of attributes that roles may have that define a list of DN's of people who are in the role.</p> <p>These values are cross referenced with the active user to determine the user's role list. One example of the use of this property is when using LDAP groups as placeholders for roles. This property only has a default value when the Netscape server type is chosen.</p>
RoleNameAttribute	cn	<p>The attribute for retrieved roles that is the common name of the role. If this value is "dn" it is interpreted specially as the entire dn of the role as the role name.</p>
RoleScope	onelevel	<p>The role search scope. The supported values for this are:</p> <ul style="list-style-type: none"> • onellevel • subtree <p>If you do not specify a value or if you specify an invalid value, the default value is used.</p>

Property	Default Value	Description
UserRoleMembershipAttributes	For iPlanet/SunONE: nsRoleDN For ActiveDirectory: memberOf For all others: none	The user's role membership attributes property is used to define an attribute that a user has that contains the DN's of all of the roles as user is a member of. These comma-delimited values are then cross-referenced with the roles retrieved in the role search base and search filter to come up with a list of user's roles.
UserFreeformRoleMembershipAttributes	None	The "freeform" role membership attribute list. Users who have attributes in this comma-delimited list are automatically granted access to roles whose names are equal to the attribute value. For example, if the value of this property is "department" and user's LDAP record has the following values for the department attribute, { "sales", "consulting" }, then the user will be granted roles whose names are "sales" and "consulting".
Referral	ignore	The behavior when a referral is encountered. The valid values are those dictated by LdapContext, for example, "follow", "ignore", "throw".
DigestMD5AuthenticationFormat	DN For OpenLDAP: User-name	The DIGEST-MD5 bind authentication identity format.
UseUserAccountControlAttribute	For most LDAP servers: false For ActiveDirectory: true	The UserAccountControl attribute to be used for detecting disabled user accounts, account expirations, password expirations and so on. ActiveDirectory also uses this attribute to store the above information.
controlFlag	optional	Indicates whether authentication with this login module is sufficient to allow the user to log in, or whether the user must also be authenticated with another login module. Rarely set to anything other than "sufficient" for any login module. Note: controlFlag is a generic login module option rather than an LDAP configuration property.

LDAP Security Provider

(Not applicable to Online Data Proxy) The LDAP security provider includes authentication, attribution, and authorization providers. Add an LDAP provider to a security configuration to authenticate administrator logins (on the "admin" security configuration on the "default" domain) or device user logins (any custom security configuration for that purpose).

You can configure these providers:

- **LDAPLoginModule**– provides authentication services. Through appropriate configuration, you can enable certificate authentication in **LDAPLoginModule**.
- (Optional)**LDAPAuthorizer** or **RoleCheckAuthorizer** – provide authorization services for **LDAPLoginModule**. **LDAPLoginModule** works with either authorizer. In most production deployments, you must always configure your own authorizer. However, if you are authenticating against a service other than LDAP, but want to perform authorization against LDAP, you can use the **LDAPAuthorizer**.

The **RoleCheckAuthorizer** is used with every security configuration but does not appear in Sybase Control Center.

Use **LDAPAuthorizer** only when **LDAPLoginModule** is not used to perform authentication, but roles are still required to perform authorization checks against the LDAP data store. If you use **LDAPAuthorizer**, always explicitly configure properties; for it cannot share the configuration options specified for the **LDAPLoginModule**.

You need not enable all LDAP providers. You can also implement some LDAP providers with providers of other types. If you are stacking multiple LDAP providers, be aware of, and understand the configuration implications.

Active Directory Considerations

If you are using Active Directory as a security provider, ensure you understand the implications of using this LDAP directory in a production environment of Unwired Platform.

Consider these design implications when you extend Active Directory security to Unwired Platform production environments:

- Shared identities among Sybase components – if you are using Active Directory for all authentication requests (that is, administration logins to access Sybase Control Center and application logins to access data), you can also set up Sybase Control Center to use this Active Directory repository. This allows users to have a shared user identity in both components, but Sybase cautions users that this can complicate deployment of the platform. For example, the user identity must be already configured both as an Unwired Platform user as well as an administrator.
- LDAP data structure and administrative control – Unwired Platform requires that roles be added to the repository in specific locations, and therefore, that security configurations also correctly point to the location of the roles. Otherwise, authentication may be problematic. Coordinate the implementation of these roles with the LDAP administrator.

Configuration Best Practices for Multiple LDAP Trees

Use the Unwired Platform administration perspective to configure LDAP authentication and authorization security providers, which are used to locate LDAP user information when organizational user groups exist within multiple LDAP trees.

To accommodate an LDAP tree structure that cannot be directly accessed using one search base:

- Create an LDAP authentication module for each level in the hierarchy – during the authentication process, Unwired Platform tries to authenticate against every login module in the ordered list until authentication succeeds or until it reaches the end of the list. Depending on the number of login modules you configure, this approach may have some performance issues.
- Use different AuthenticationScopes for performing user searches – specify the root node of a particular LDAP tree, by entering `AuthenticationSearchBase="dc=sybase, dc=com"` and set `Scope=subtree`. Unwired Platform performs an LDAP query against the entire subtree for authentication and authorization information. Depending on the number of AuthenticationScope within the LDAP tree structure, this approach can have performance implications.
- If multiple servers are clustered together to form a large logical directory tree, configure the LDAPLoginModule by setting the `Referral` property to `follow`.
- If subjects have been made members of too many LDAP groups and the search for physical roles results in too many results, the maximum result limit may be reached and authentication fails. To avoid this, narrow the `RoleSearchBase` to LDAP groups that are relevant only to Unwired Platform.

LDAP Role Computation

Role checks are the primary means of performing access control when using LDAP authentication. Authentication and attribution capabilities both utilize role computation techniques to enumerate roles that both authenticated users have.

There are three distinct types of role constructs supported by LDAP providers; each may be used independently, or all three may be configured to be used at the same time.

- User-level role attributes, specified by the `UserRoleMembershipAttributes` configuration property, are the most efficient role definition format. A user's roles are enumerated by a read-only directory server-managed attribute on the user's LDAP record. The advantage to this technique is the efficiency with which role memberships can be queried, and the ease of management using the native LDAP server's management tools. These constructs are supported directly by ActiveDirectory products, and use these configuration options:
 - `UserRoleMembershipAttributes` – the multivalued attribute on the user's LDAP record that lists the role DNs that the user is a member of. An example value for this property is "memberOf" on ActiveDirectory.

- **RoleSearchBase** – the search base under which all user roles are found, for example, "ou=Roles,dc=sybase,dc=com". This value may also be the root search base of the directory server.
- **RoleFilter** – the search filter that, coupled with the search base, retrieves all roles on the server.
- (Optional) **RoleScope** – enables role retrieval from subcontexts under the search base.
- (Optional) **RoleNameAttribute** – choose an attribute other than "cn" to define the name of roles.

These properties retrieve correct values automatically based on the type of server you configure.

- LDAP group role definitions may be used as role definitions. This is a common construct among older LDAP servers, but is supported by nearly all LDAP servers. You may select this approach to use the same LDAP schema across multiple LDAP server types. Unlike the user-level role attributes, LDAP group memberships are stored and checked on a group-by-group basis. Each defined group, typically of objectclass "groupofnames" or "groupofuniquenames," has an attribute listing all of the members of the group. The configuration settings used are the same as for user-level role attributes, except for the **RoleMemberAttributes** property, which replaces the **UserRoleMembershipAttributes** property. This property defines a comma-delimited list of attributes that contain the members of the group. An example value for this property is "uniquemember,member," which represents the membership attributes in the above-mentioned LDAP objectclasses.
- Freeform role definitions are unique in that the role itself does not have an actual entry in the LDAP database. A freeform role starts with the definition of one or more user-level attributes. When roles are calculated for a user, the collective values of the attributes (each of which may be multivalued) are added as roles to which the user belongs. This technique is particularly useful when the overhead of managing roles uses are administration-heavy. For example, assign a freeform role definition that is equivalent to the department number of the user. A role check performed on a specific department number is satisfied only by users who have the appropriate department number attribute value. The only property that is required or used for this role mapping technique is the comma-delimited **UserFreeformRoleMembershipAttributes** property.

LDAP Login and Authorization Modules

LDAP login and authorization modules can sometimes share a common configuration. However, authorizers do not inherit configuration from login modules you configure. Configurations must be explicit.

In the case where both **LDAPLoginModule** and **LDAPAuthorizer** are configured:

- **Matching configuration** – **LDAPAuthorizer** simply skips the role retrieval.
- **Differing configuration** – **LDAPAuthorizer** proceeds with the role retrieval from the configured backend, and performs the authorization checks using the complete list of roles (from both the login module and itself). Even in the case where multiple

LDAPLoginModules are configured, only one LDAPAuthorizer is required as it compares its configuration with the configuration used for the successful authentication of the user.

Certificate Configuration Properties

The certificate validation provider contributes only authentication services. For Unwired Server security, configure these properties from the corresponding tab in the Security node of Sybase Control Center.

To configure certificate validation with another security provider, ensure you configure the certificate validation properties before other login modules that will support this validation service.

Note: This provider cannot be used for administrative security (in the "admin" security configuration).

Table 10. Authentication properties

Property	Default value	Description
validatedCertificateIsIdentity	false	Specifies if the certificate should be set as the ID for the authenticated subject. This option should be set to false if the CertificateValidationLoginModule is used in conjunction with other login modules that establish user identity based on the validated certificate.

NTProxy Configuration Properties

(Not applicable to Online Data Proxy) Configure these properties to allow the operating system's security mechanisms to validate user credentials using NTProxy (Windows Native OS). Access these properties from the Authentication tab of the Security node in Sybase Control Center.

Table 11. Authentication properties

Properties	Default Value	Description
Extract Domain From Username	true	If set to true, the user name can contain the domain in the form of <username>@<domain>. If set to false, the default domain (described below) is always used, and the supplied user name is sent to through SSPI untouched.

Properties	Default Value	Description
Default Domain	The domain for the host computer of the Java Virtual Machine.	Specifies the default host name, if not overridden by the a specific user name domain.
Default Authentication Server	The authentication server for the host computer of the Java Virtual Machine.	The default authentication server from which group memberships are extracted. This can be automatically determined from the local machine environment, but this property to bypass the detection step.
useFirstPass	false	If set to true, the login module attempts to retrieve only the user name and password from the shared context. It never calls the callback handler.
tryFirstPass	false	If set to true, the login module first attempts to retrieve the user name and password from the shared context before attempting the callback handler.
clearPass	false	If set to true, the login module clears the user name and password in the shared context when calling either commit or abort.
storePass	false	If set to true, the login module stores the user name and password in the shared context after successfully authenticating.

NTProxy Security Provider

(Not applicable to Online Data Proxy) NTProxy — sometimes known as native Windows login — is an Unwired Server provider that integrates with existing Windows login security mechanisms. Add an LDAP provider to a security configuration to authenticate administrator logins (on the "admin" security configuration on the "default" domain) or device user logins (any custom security configuration for that purpose).

If added to a particular security configuration, users or administrators can authenticate with their native Windows user name and password, which gives them access to roles that are based on their existing Windows memberships.

The NTProxy provider fulfills authentication services only with classes in `csi-nativeos.jar`; role-based access control and attribution are not directly supported.

Groups are also not supported in NTProxy. Instead, group memberships are transformed into a role of the same name and can be mapped in Sybase Control Center.

HTTP Basic Authentication Properties

The `HttpAuthenticationLoginModule` provider authenticates the user with given credentials (user name and password) against a HTTP(S) enterprise information system (EIS) using a GET against an URL that requires BASIC authentication, and can be configured to retrieve a cookie with the configured name and add it to the JAAS subject to facilitate single sign-on.

Note: If you are using this provider for the "admin" security configuration, ensure you make corresponding provider changes for Sybase Control Center logins. See *Enabling Authentication and RBAC for Administrators* in the *Security* guide.

Table 12. HttpAuthenticationLoginModule properties

Property	Description
URL	The HTTP(S) URL that authenticates the user. For SSO, this is the server URL from which Unwired Server acquires the SSO cookie/token.
Disable certificate validation	(Optional) – the default is <code>False</code> . If set to <code>True</code> , disables certificate validation when establishing an HTTPS connection to the EIS using the configured URL. Set to <code>True</code> only for configuration debugging.
SSO cookie name	(Optional) – name of the cookie that is set in the session and holds the SSO token for single sign-on. The authentication provider ignores the status code when a SSO cookie is found in the session. If the cookie is found, authentication succeeds regardless of the return status code.
Roles HTTP header	(Optional) – comma separated list of roles granted to the authenticated user for role-based authorization.
Successful connection status code	HTTP status code interpreted as success when connection is established to the EIS. The default is 200.
HTTP connection timeout interval	The value, in seconds, after which an HTTP(s) connection request to the EIS times out. If the HTTP connection made in this module (for either user authentication or configuration validation) does not have a time out set, and attempts to connect to an EIS that is unresponsive, the connection hangs, which could potentially cause Unwired Server to hang. Setting the timeout interval ensures authentication failure is reported without waiting for ever for the server to respond.

HTTP Basic Security Provider

Use an HTTP Basic provider to enable automatic application registration. This provider is required when registration is set to automatic, and will be used for integrating with third-party security providers like SiteMinder.

A `LoginModule` validates standard username/password style credentials by passing them to a Web server. Configure the `URL` property to point to a Web server that challenges for Basic authentication.

Best practice guidelines include:

- Using an HTTPS URL to avoid exposing credentials.
- If the Web server's certificate is not signed by a well known CA, import the CA certificate used to sign the Web server's certificate into the Unwired Server `truststore.jks`. The truststore is prepopulated with CA certificates from reputable CAs.
- If this Web server returns a cookie as part of successful authentication, set the `SSO Cookie Name` configuration property to the name of this cookie. Upon successful authentication, this login module places the cookie value into an `HttpSSOTokenCredential` object and attaches it to the `java.security.Subject` as a public credential.

Note: The HTTP Basic login module is the module that can either be used for SSO tokens or HTTP basic without SSO. The sole condition being that the backend support HTTP Basic authentication.

- When using this module in lieu of the deprecated `SAPSSOTokenLoginModule`, the cookie name is typically "MYSAPSSO2".

For example, SiteMinder is often used in mobile deployments to protect existing Web-based applications. Existing users point their browser at a URL, and SiteMinder intercepts an unauthenticated session to challenge for credentials (Basic). When the authentication succeeds, it returns a `SMSESSION` cookie with a Base64-encoded value that can be used for SSO into other SiteMinder enabled systems.

SAP SSO Token Authentication Properties

The `SAPSSOTokenLoginModule` has been deprecated, Use the `HttpAuthenticationLoginModule` when SAP SSO2 token authentication is required. This authentication module will be removed in a future release.

Table 13. SAPSSOTokenLoginModule properties

Property	Description
Implementation class	(Required) – the fully qualified class that implements the login module. <code>com.sybase.security.sap.SAPSSOTokenLoginModule</code> is the default class.
Provider type	(Required and read-only) – <code>LoginModule</code> is the only supported value.
Control flag	(Required) – <code>optional</code> is the default value. Determines how success or failure of this module affects the overall authentication decision.
SAP server URL	<p>(Required) – the SAP server URL that provides the SSO2 token. This may or may not be the same server that authenticates the user. If providing and authenticating servers are different, you must import the SAP Token provider server certificate or one of its CA signers into the Unwired Server truststore in addition to that of the authenticating server to enable HTTPS communication. In environments where the servers are different, the basic flow is:</p> <ol style="list-style-type: none"> 1. Unwired Server passes credentials over HTTPS to the token granting service. 2. An SSO2Token cookie is returned to Unwired Server. 3. The SSO2Token flows to the authenticating server, which could be an SAP EIS or a server that hosts a Web service bound to SAP function modules. <p>Note: The SAP Server URL must be configured to require BASIC authentication, not just FORM based authentication.</p>
Clear password	(Optional) – if set to <code>True</code> , the login module clears the username and password in the shared context.

Property	Description
Disable server certificate validation	(Optional) – the default is <code>False</code> . If set to <code>True</code> , disables certificate validation when establishing an HTTPS connection to the SAP server using the configured URL. Set to <code>True</code> only for configuration debugging.
SAP server certificate	(Optional) – name of the file containing the SAP certificate's public key in <code>.pse</code> format. This is required only when token caching is enabled by setting a SAP SSO token persistence data store value.
SAP server certificate password	(Optional) – password used to access the SAP server certificate.
SAP SSO token persistence data store	<p>(Optional) – JNDI name used to look-up the data source to persist the retrieved SSO2 tokens.</p> <p>Set to <code>"jdbc/default"</code> to store tokens in the Unwired Server CDB. If unconfigured, some caching is still done based on the <code>"Authentication cache timeout interval"</code> property associated with the security configuration setting.</p> <p>If you use the default setting, you do not need to set SAP SSO token persistence data store, SAP server certificate, SAP server certificate password, or Token expiration interval properties.</p> <p>To enable token caching through the <code>SAPSSOTokenLoginModule</code>:</p> <ol style="list-style-type: none"> 1. Set the SAP SSO token persistence data store value to <code>"jdbc/default."</code> 2. Download and install the SAP SSO2 token files. See <i>Installing the SAP SSO2Token Files on Unwired Server Hosts</i> in the <i>Security</i> guide. 3. Specify the correct value for the SAP server certificate, SAP server certificate, SAP server certificate password and Token expiration interval properties.
Store password	(Optional) – if set to <code>true</code> , the login module stores the username/password in the shared context after successfully authenticating the user.

Property	Description
Token expiration interval	<p>(Optional) – this property is ignored when the SAP SSO token persistence data store property is not configured. It specifies the token validity period, after which time a new token is retrieved from the SAP EIS. The default value is 120 seconds.</p> <p>Keep in mind that:</p> <ul style="list-style-type: none"> • The "Token expiration interval" cannot exceed the "Token validity period", which is the amount of time defined in the back-end SAP server for which the token is valid. • The "Authentication cache timeout" property must be less than the "Token expiration interval" property value.
Try first password	(Optional) – if set to <code>True</code> , the login module attempts to retrieve the username/password from the shared context, before calling the callback handler.
Use first password	(Optional) – if set to <code>True</code> , the login module attempts to retrieve the username/password only from the shared context, and never calls the callback handler.
HTTP connection timeout interval	The value, in seconds, after which an HTTP(s) connection request to the EIS times out. If the HTTP connection made in this module (for either user authentication or configuration validation) does not have a time out set, and attempts to connect to an EIS that is unresponsive, the connection hangs, which could potentially cause Unwired Server to hang. Setting the timeout interval ensures authentication failure is reported without waiting for ever for the server to respond.

SAP SSO Token Security Provider

The `SAPSSOTokenLoginModule` has been deprecated and will be removed in a future release. Use `HttpAuthenticationLoginModule` for SAP SSO2 token authentication.

Use `HttpAuthenticationLoginModule` for both JCo and DOE-C connections to the SAP system. Unwired Server does not provide authorization control or role mappings for user authorization; enforce any access control policies in the SAP system.

Preconfigured User Authentication Properties

The PreConfiguredUserLoginModule authenticates the Unwired Platform Administrator user whose credentials are specified during installations.

This login module is recommended only to give the Platform administrator access to Sybase Control Center so it can be configured for production use. Administrators are expected to replace this login module immediately upon logging in for the first time. For details on how to setup administrator authentication in a production deployment, see *Enabling Authentication and RBAC for Administrator Logins* in the *Security* guide.

The PreConfiguredUserLoginModule:

- Provides role based authorization by configuring the provider `com.sybase.security.core.RoleCheckAuthorizer` in conjunction with this authentication provider.
- Authenticates the user by comparing the specified username/password against the configured user. Upon successful authentication, the configured roles are added as Principals to the Subject.

Table 14. PreConfiguredUserLoginModule properties

Property	Description
User name	A valid user name.
Password	The encoded password hash value.
Roles	<p>Comma separated list of roles granted to the authenticated user for role-based authorization. Platform roles include "SUP Administrator" and "SUP Domain Administrator".</p> <p>Roles are mandatory for "admin" security configuration. For example, if you define "SUP Administrator" to this property, the login id in the created login module has Platform administrator privileges.</p> <hr/> <p>Note: If you use other values, ensure you map Unwired Platform roles to the one you define here.</p>

Preconfigured User Login Security Provider

Preconfigured login is configured to authenticate the supAdmin user with a password that was defined when Unwired Platform was installed. Therefore, an administrator must use `supAdmin` with `<supAdminPwd>` when initially logging in to Sybase Control Center for the first time.

Note: Do not forget this installer-defined password. The installer hashes the password with a SHA-256 algorithm before it is saved as part of the PreconfiguredLoginModule configuration, and it cannot be returned to clear text once it is hashed.

Once logged in, the Unwired Platform administrator immediately reconfigures the "admin" security configuration to replace this provider with a production-grade security provider like LDAP. If you configure a new provider in Unwired Platform and Sybase Control Center and login fails, review possible login failure solutions in the *Troubleshooting* guide.

Certificate Authentication Properties

Add and configure authentication provider properties for CertificateAuthenticationLoginModule, or accept the default settings.

Note: This provider cannot be used for administrative security (in the "admin" security configuration).

Table 15. CertificateAuthenticationLoginModule properties

Property	Description
Implementation class	The fully qualified class that implements the login module. <code>com.sybase.security.core.CertificateAuthenticationLoginModule</code> is the default class.
Provider type	<code>LoginModule</code> is the only supported value.
Control flag	Determines how success or failure of this module affects the overall authentication decision. <code>optional</code> is the default value.
Clear password	(Optional) If true, the login module clears the user name and password from the shared context. The default is false.
Store password	(Optional) If true, the login module stores the user name and password in the shared context. The default is false.
Try first password	(Optional) If true, the login module attempts to retrieve user name and password information from the shared context, before using the callback handler. The default is false.
Use first password	(Optional) If true, the login module attempts to retrieve the user name and password only from the shared context. The default is false.
Enable revocation checking	(Optional) Enables online certificate status protocol (OCSP) certificate checking for user authentication. If you enable this option, you must enable OCSP in Unwired Server. This provider uses the values defined as part of the SSL security profile. Revoked certificates result in authentication failure when both of these conditions are met: <ul style="list-style-type: none"> • revocation checking is enabled • OCSP properties are configured correctly

Property	Description
Regex for username certificate match	<p>(Optional) By default, this value matches that of the certificates common name (CN) property used to identify the user.</p> <p>If a mobile application user supplies a user name that does not match this value, authentication fails.</p>
Trusted certificate store	<p>(Optional) The file containing the trusted CA certificates (import the issuer certificate into this certificate store). Use this property and <code>Store Password</code> property to keep the module out of the system trust store.</p> <p>The default Unwired Server system trust store is <code><Unwired-Platform_InstallDir\Servers\Unwired-Server\Repository\Securitytruststore\truststore.jks</code>.</p> <hr/> <p>Note: This property is required only if <code>Validate certificate path</code> is set to true.</p>
Trusted certificate store password	<p>(Optional) The password required to access the trusted certificate store. For example, import the issuer of the certificate you are trying to authenticate into the shared JDK cacerts file and specify the password using this property.</p> <hr/> <p>Note: This property is required only if <code>Validate certificate path</code> is set to true.</p>
Trusted certificate store provider	<p>(Optional) The keystore provider. For example, "SunJCE."</p> <hr/> <p>Note: This property is required only if <code>Validate certificate path</code> is set to true.</p>
Trusted certificate store type	<p>(Optional) The type of certificate store. For example, "JKS."</p> <hr/> <p>Note: This property is required only if <code>Validate certificate path</code> is set to true.</p>

Property	Description
Validate certificate path	If true (the default), performs certificate chain validation of the certificate being authenticated, starting with the certificate being validated. Verifies that the issuer of that certificate is valid and is issued by a trusted certificate authority (CA), if not, it looks up the issuer of that certificate in turn and verifies it is valid and is issued by a trusted CA. In other words, it builds up the path to a CA that is in the trusted certificate store. If the trusted store does not contain any of the issuers in the certificate chain, then path validation fails. For information about adding a certificate to the truststore, click <i>Preparing Certificates and Key Pairs</i> .

Certificate Security Provider

Use the Unwired Server CertificateAuthenticationLoginModule authentication provider to implement SSO with an SAP enterprise information system (EIS) with X.509 certificates.

Unwired Server does not provide authorization control or role mappings for user authorization; enforce any access control policies in the EIS.

Roles and Mappings

Role mapping occurs when an administrator maps logical roles to physical roles using Sybase Control Center as part of a security configuration or a deployment package. The physical roles are the roles and groups in the underlying security repository. The mapped role determines the security role requirement for a user at runtime to access a resource that is using the security configuration on which the mapping is defined.

In Unwired Platform, the mapped role determines what security roles apply to users when they attempt to perform an operation from the mobile application (device users) or Sybase Control Center (administrators).

Role mappings are defined as part of a security configuration that you can assign to a particular domain. Administrators can assign the same security configuration to multiple domains; ensure that these mappings are suitable for all domains to which the security configuration is assigned. Consider an example where security configuration is shared between domainA and domainB.

1. The platform administrator (the administrator assigned the SUP administration role) creates a security configuration called AllDomains.
2. The platform administrator assigns the AllDomain to the domain, and maps the EmpRole role to SalesGroupRole in the security repository used by that configuration.

This change that is specific to just domainA is also implemented in domainB even though the domain administrator of domainB did not explicitly make, or require, the change. But the role mapping is propagated to domainB as well. To avoid this, the Unwired Platform administrator may want to create multiple security configurations so that underlying mechanisms can stay the same, but specific role mappings can be made for each.

For device user security, there is an increased flexibility for packages as they are deployed. If a security configuration is inappropriate, or if a role is not mapped at all that is used by the package, the platform or domain administrator can override or extend the role mappings defined for the security configuration. Package-level role mappings always take precedence in such a scenario.

Assigning a Security Configuration to a Domain

Assign security configurations to one or more domains. This allows the supAdmin to offer a security repository for application user authentication and authorization, as well as to share security providers across domains in case one tenant uses multiple domains.

Prerequisites

A supAdmin must already have created one or more security configurations in the Security node of Sybase Control Center.

Task

1. In the left navigation pane, navigate to *ClusterName* > **Domains** > *DomainName* > **Security**
2. In the right administration pane, select the **Security Configurations** tab and click **Assign**.
3. From the list of available security configurations, select the appropriate configuration for domain security, and click **OK**.
If successful, an Assigned successfully message appears, and the newly added security configuration is listed in the domain-level Security node.

Applications

An Application is the runtime entity that can be directly correlated to a native or mobile workflow application. The application definition on the server establishes the relationship among packages used in the application, domain that the application is deployed to, user activation method for the application, and other application specific settings.

- For native replication/messaging applications, one or more MBO packages can be assigned to an application. If the application developer uses the same package in a different application, the MBO package must be assigned to that application.
- For the mobile workflow applications, all MBO packages are accessible in all domains, so no MBO packages need to be assigned to the mobile workflow application (HWC).
- For Online Data Proxy applications, no MBO package assignments are needed as well.

Applications are managed and monitored by administrators on the Sybase Control Center. They are created automatically or manually through Sybase Control Center.

An application ID uniquely identifies an application to Unwired Server. Application connection templates enable administrators to manually register application connections in

Unwired Server with predefined settings. Templates also enable automatic activation of devices (described later). Users are associated with one or more applications through application connections. Administrators can view application users in Sybase Control Center as soon as a user logs onto the application from a device.

Setting Up Application and User Connections

The basic steps for setting up application and user connections involve creating an application, and registering application connections to associate users to the application.

Application Creation

There are two ways an application gets created - automatic and manual.

Automatic Application Creation

Applications are created automatically, when the system administrator deploys a package. The mobile workflow application (HWC) is created during Unwired Server installation.

An application is created automatically when an MBO package is deployed to the server. In case of upgrade from a previous server version, an application is created for all deployed MBO packages. The default name of an application is the same as the package name.

Note: An application is primarily used for tracking purpose. At this point, an application connection template is also created, but not used at this time since automatic registration is not available for native applications (MBO package client applications).

For Online Data Proxy, there is no such automatic creation of applications. Applications must be created manually.

Manually Creating Applications

Create an application manually by assigning a unique application ID and other key application properties, such as domain, MBO package, security configuration, among others. At this time, the manual process is only needed for Online Data Proxy applications or when using a Hybrid Web Container built using the iOS sample, where developers can use their own application IDs for workflow applications.

Launching the Application Creation Wizard

Use the Application Creation wizard to register an application.

1. In the left navigation pane of Sybase Control Center, click the **Applications** node and select the **Applications** tab in the right administration pane.
2. To register an application, click **New**.
The Application Creation wizard is displayed.

Setting General Application Properties

Provide general application properties such as the application ID, description, security configuration and domain details while registering the application.

1. In the Application Creation Wizard, enter a unique **Application ID**, following application ID guidelines.
2. Enter a **Display name** and **Description** for the application.
3. Select the appropriate security configuration from the **Security Configuration** drop-down list.
4. Select the appropriate domain from the **Domain** drop-down list.
5. (Optional) Select a package from the list to assign an application connection template.
 - a) Select **Configure additional settings**, and click **Next**.
 - b) In the Application Creation wizard, enter a **Template name** and **Description** for the application connection template.
 - c) To reuse the configuration of an existing template, select a **Base template** from the drop-down list.
 - d) Configure the following property categories as required.
 - Apple Push Notifications
 - Application Settings
 - BlackBerry Push Notifications
 - Connection
 - Custom Settings
 - Device Advanced
 - Device Info
 - Proxy
 - Security Settings
 - User Registration
6. Click **Finish** to register the application with the configured settings.

Application ID Overview

Applications can be directly correlated to a native application or mobile workflow container instance on device. A native application is the single binary deployed to device which may use one or more MBO packages. The mobile workflow application is a collection of workflow packages and constitutes as one application. One or more MBO packages can be assigned to an application. If application developers want to use the same package in a different application, they can do that by assigning the MBO package to that application using Sybase Control Center.

An application ID uniquely identifies the application and must be used to register an application connection and also used in the device application in case of Online Data Proxy client for activation of its application connection.

Modifying Application Properties

Associate the application with one or more domains and packages.

1. In the left navigation pane of Sybase Control Center, click the **Applications** node and select the **Applications** tab in the right administration pane.
2. Select an application listed under Application ID.
3. Click **Packages**.
4. In Review Assignment, select the domains to associate with the Application.
 - a) To see more domains, click +.
 - b) In Available Domains, select one or more domain from the list .
5. Select the packages to associate with the Application.
 - a) To see more packages, click +.
 - b) In Available Packages, select one or more package from the list of available packages.
6. Click **OK**.

Deleting Applications

Delete an application to remove all the registered users, connections, and subscriptions associated with those connections.

1. In the left navigation pane of Sybase Control Center, click the **Applications** node and select the **Applications** tab in the right administration pane.
2. Select the application and click **Delete**.

Application Connection Activation Options

An application connection can be activated from the device by either providing valid credentials, or an activation code for a pre-registered application connection. The credentials-based approach, referred to from this point on as automatic registration, relies on application connection template properties (application ID, security configuration, automatic registration enabled). The activation-based approach, referred to from this point as manual registration, relies on matching user name and activation code sent from the device to an existing application connection registered for the user.

Note: Application connection is not used for native replication applications at this time.

Information and guidelines:

- Native messaging applications can only be activated manually.

- Application template are used for automatic activation. Therefore, when setting up the application template for automatic registration, be sure to set up the security configuration, domain, the application ID, and automatic registration enabled properties in application settings. Those are used for automatic application registration.

When a client application connects to the server with its application ID and credentials, and requests automatic registration, the application ID is used to look up a matching template. If that template allows automatic registration (the Automatic Registration Enabled property is set to true), the security configuration in the template is used to validate the credentials. Upon successful validation of those credentials, the user identity is registered in the Unwired Server. The client application may also include the security configuration as part of the username (user@securityconfiguration) and in that case, the security configuration (in addition to application ID) is used to look up a matching template. If no or multiple templates are detected, the registration request fails.

- Supported device client activation options:

Device Client Type	Automatic Registration	Manual Registration
Workflow messaging	X	X
Native messaging		X
Online Data Proxy	X	X

Managing and Searching for Applications

You can view the applications registered through the Sybase Control Center

1. In the left navigation pane, click the **Applications** node.
2. In the right administration pane, click the **Application** tab.
You can view the list of registered applications.

Viewing Assigned Connections

View the properties of the connections assigned to an application.

1. In the right administration pane, select the **Applications** tab.
2. Select the application from the list.
3. Click **Application Connections**.
4. Click **Refresh** to refresh the list that displays the application connections.

Viewing Assigned Application Users

View the list of the users assigned to an application

1. In the right administration pane, select the **Applications** tab.
2. Select the application from the list.

3. Click **Application Users**.
4. Click **Refresh** to refresh the list that displays the application users.

Viewing Correlated Application Details

Select one or more applications, then view correlated application details in several categories, including packages, application users, and application connections.

1. In the left navigation pane, click the **Applications** node.
2. In the right administration pane, click the **Application** tab.
You can view the list of registered applications.
3. Select one or more applications from the list.
4. Select one of the buttons to view related packages, application users, or application connections.
 - **Packages** – view correlated domains and packages.
 - **Application Users** – view correlated applications.
 - **Application Connections** – view correlated applications.
5. In Review Assignment, check the information.
6. Click **OK**.

Refreshing the Application View

Refresh the list of all available applications registered through the Sybase Control Center.

1. In the right administration pane, select the **Applications** tab.
2. To view the list of registered applications, click **Refresh**.

Searching for Applications

Search for registered applications from the default view, or perform an advanced search. The advanced search enables you to search through applications, users, application connections, packages, and subscriptions, filtering out results at each level until you obtain very specific results.

Searching from the Default View

Search for applications that are registered in the Sybase Control Center.

1. In the right administrations pane, select the **Applications** node.
2. To set the search criteria, select the criteria from the **Search** drop-down list.
3. Add a search string.
4. Click **Go**.
All the applications that match the search criteria provided are populated in the table.

Performing an Advanced Search

search through applications, users, application connections, packages, and subscriptions, filtering out results at each level until you obtain very specific results.

1. In the right administrations pane, select the **Applications** node.
2. Select **Advanced Search**.
3. In Advanced Search, enter an application ID, then click **Go**, or click **Go** to display a list of applications.
4. Select an application, and click **Search**.
The **Users**, **Application Connections**, **Packages**, and **Subscriptions** tabs remain.
5. Select one of the remaining tabs, select the search criteria, and click **Search**.
6. Continue this process until you have the information you seek. Each time you search in a one of the categories, the tab is removed. In this way you refine your search.

Application Users

In Unwired Platform, an application user is an identity registered as the user of one or more versions of a device application. Application users are managed in Sybase Control Center.

For native replication applications, an application user is automatically registered upon first successful authentication using the security configuration associated with the package that user is trying to access. For native messaging applications, an application user is automatically registered upon first successful synchronization after manual registration. For workflow and Online Data proxy applications, the application user is registered upon successful registration whether manual or automatic. A user can have multiple devices.

The platform administrator can view the user name and the security configuration that was used to authenticate the user. In addition, the administrator can remove users that no longer exist.

Note: SAP DOE-C package users are not registered in Unwired Server. Those users are authenticated by their respective DOE back-end servers.

Deleting Application Users

Delete a user to remove the entry as well as personalization data from the cache database.

1. In the left navigation pane of Sybase Control Center, click the **Applications** node and select the **Application Users** tab in the right administration pane.
2. Select the user and click **Delete**.

The user entry and data are removed.

Checking Application User Assignments

Check which applications are used by registered users.

1. In the left navigation pane of Sybase Control Center, click the **Applications** node and select the **Application Users** tab in the right administration pane.
2. Select an application user and click **Applications**.

All applications used by the user are listed in the dialog.

Searching for Application Users

Search for application users according to the criteria you specify.

1. In the right administration pane, select the **Applications Users** node.
2. Choose the criteria from the **Search** drop-down list for which you want to search for the required user.
3. Click **Go**.

The user information is populated according to the criteria you have specified.

Refreshing the Application Users View

Refresh the application user list to display current information about registered users.

1. In the right administration pane, select the **Applications** node, then the **Application Users** tab.
2. Click **Refresh** to view the current information of all users.

Application Connections

Application connections associate an application instance with a user. An application may be used by many users, and a user may be associated with many applications.

Application ID Guidelines

Follow these guidelines for choosing an appropriate application ID while registering application connection for use by native messaging, workflow client, or Online Data Proxy. Failure to specify the correct application ID would result in failure when the client tries to activate itself even though the user name and activation code do match.

Registration Type	Application ID Guidelines
Native messaging client	The application ID should always be left empty. If using a preexisting registration template (such as Default), then the application id would be empty already.

Registration Type	Application ID Guidelines
Native Android client application	An application ID is required. Register the application connection using the template created for the application.
Workflow client	<ul style="list-style-type: none"> • 2.0.1 or earlier version – leave the application ID empty. • 2.1 or later version – use preexisting "HWC" template, or, if using your own template, make sure that "HWC" is set as the application ID in the template. • iOS Sample container 2.1 version – use the template you have created. The application ID used by the iOS sample container should match the application ID specified in registration.
Online Data Proxy client	Register application connection using the template created for the application. Existing templates can be found in the Applications > Application Connection Template tab.

Registering or Reregistering Application Connections

Use Sybase Control Center to trigger the registration of an application connection, or reregister an application connection when the activation code has expired.

1. In the left navigation pane, click the **Applications** node.
2. In the right administration pane, click the **Application Connections** tab.
3. Choose an action:
 - Click **Register** to register a new application connection. Using the Activation Code, this application is then paired with a user and a device.
 - Click **Reregister** to associate the application with a new device and user pairing. For example, reregister the application connection if someone loses their device. By reregistering the application connection, the user then receives the same applications and workflows as the previous device.
4. In the Register Application Connection or the Reregister Application Connection dialog.
 - a) For new device registration only, type the name of the user that will activate and register the device. For reregistrations or clones, the same name is used and cannot be changed.
 - b) Select the name of the template for initial application connection registration. The template you use supplies initial values in the subsequent fields.
 - Default – a default template that you can use as is, or customize.
 - HWC – a default template for mobile workflows (containers). Use as is, or customize. If you use the HWC template, Application ID must be set to HWC.

- Custom - customized templates are listed.
5. Change the default field values for the template you have chosen. If you are using the default template, you must provide the server name.
If you are using Relay Server, ensure the correct values are used.
 - **Server name**- the DNS name or IP address of the primary Unwired Server, such as "myserver.mycompany.com". If using Relay Server, the server name is the IP address or fully qualified name of the Relay Server host.
 - **Port**- the port used for messaging connections between the device and Unwired Server. If using relay server, this is the Relay Server port. Default: 5001.
 - **Farm ID**- a string associated with the relay server farm ID. Can contain only letter A-Z (uppercase or lowercase), numbers 0-9, or a combination of both. Default: 0.
 - **Application ID**- the application ID registered for the application. The value differs according to application client type - native messaging, workflow, or Online Data Proxy client. See *Application ID Overview* for guidelines.
 - **Security Configuration**- select the security configuration relevant for the application connection.
 - **Domain**- select the domain for which you want to register the application connection with.
 - **Activation code length** - the number of characters in the activation code. If you are reregistering or cloning a device, this value cannot be changed.
 - **Activation expiration**- the number of hours the activation code is valid.
 6. (Optional) Select the check box adjacent to **Specify activation code** to enter the code sent to the user in the activation e-mail. This value can contain letter A - Z (uppercase or lowercase), numbers 0 - 9, or a combination of both. Acceptable range: 1 to 10 characters.
 7. Click **OK**

Searching for Application Connections

Set search criteria to filter connections viewed in the Application Connections tab

1. In the right navigation pane, click the **Applications** node.
2. In the right administration pane, click the **Application Connections** tab.
3. To set the search criteria, configure these search elements:
 - Choose the connection information column name you want to enter a search value for.
 - Type or choose the value for the column name you selected.

Deleting Application Connections

Delete an application connection to remove a user assignment to an application connection.

1. In the left navigation pane of Sybase Control Center, click the **Applications** node and select the **Application Connections** tab in the right administration pane.

2. Select the application connection and click **Delete**.

Editing the Application Connection Properties

Modify or update the properties of an application connections

1. In the left navigation pane, click the **Applications** node.
2. In the right administration pane, click the **Applicaton Connections** node.
3. Select an application connection from the list.
4. Click **Properties**.
 - a) In the Application Connection Properties dialog, select the category from the left pane.
 - b) Update or modify the property and its value.
 - c) Click **OK**.

Note: When the application end-point for a registered application is modified under the **Proxy** property, you have to manually update the **Address** in the proxy properties of the connection pool.

Cloning Application Connections

Create a duplicate copy of an application connection configuration settings. This allows you to retain user information and pair it with a different device in the event that a user gets a new or alternate device.

1. In the left navigation pane, click the **Applications** node.
2. In the right administration pane, click the **Application Connections** node.
3. Check the box adjacent to the connection you want to clone and click **Clone**.
4. Edit the configuration settings associated with the application connection.
 - **Server name**- the DNS name or IP address of the primary Unwired Server, such as "myserver.mycompany.com". If using relay server, the server name is the IP address or fully qualified name of the relay server host.
 - **Port**- the port used for messaging connections between the device and Unwired Server. If using relay server, this is the relay server port. Default: 5001.
 - **Farm ID**- a string associated with the relay server farm ID. Can contain only letter A-Z (uppercase or lowercase), numbers 0-9, or a combination of both. Default: 0.
 - **Application ID**- the application ID registered for the application.
 - **Security Configuration**- select the security configuration relevant for the application connection.
 - **Domain**- select the domain for which you want to register the application connection with.
 - **Activation expiration**- the number of hours the activation code is valid.

5. (Optional) Select the check box adjacent to **Specify activation code** to enter the code sent to the user in the activation e-mail. This value can contain letter A - Z (uppercase or lowercase), numbers 0 - 9, or a combination of both. Acceptable range: 1 to 10 characters.
6. Click **OK**

Tracing Application Connections

Send a request to Unwired Server to retrieve log files for an application connection.

1. In the left navigation pane, select the **Applications** node.
2. In the right administration pane, click **Application Connections** tab.
3. Select an application connection, and click **Get Trace**.
The application connection status must be "online" to retrieve the logs.
4. Click **OK**.
5. When the application connection is online, check the application connection log. The default location for single node and cluster installations is
`<UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers
\MessagingServer\Bin\ClientTrace .`

Locking and Unlocking Application Connections

Lock or unlock connections to control which users are allowed to synchronize data. Locking an application connection is an effective way to disable a specific user without making changes to the security profile configuration to which he or she belongs.

1. In the left navigation pane, select the **Applications** node.
2. In the right administration pane, select the **Application Connections** tab.
3. Select the application connection you want to manage, and:
 - If the connection is currently unlocked and you want to disable synchronization, click **Lock**.
 - If the connection is currently locked and you want to enable synchronization, click **Unlock**.
4. In the confirmation dialog, click **OK**.

Application Connection Templates

An application connection template is a model or pattern used to standardize connection properties and values for a specific application connections so that they can be reused. A template allows you to quickly create actual application connections.

Two default templates are available: Default and HWC. The Default is for registering an application connection without an application ID (backward compatibility scenario, as well as for 2.1.1 native messaging clients). The HWC template is for registering application connections for 2.1 or later version of Hybrid Web Container clients.

Creating Application Connection Templates

Create application connection templates by setting appropriate properties and values.

1. In the left navigation pane, click the **Applications** node.
2. In the right administration pane, click the **Application Connection Templates** tab.
3. Click **New**.
4. Enter the **Template name** and **Description** for the application connection template.
5. Select the **Base template** from the drop-down list.
6. You can configure any of the following profiles. See *Application Settings*:
 - Apple Push Notifications
 - Application Settings
 - BlackBerry Push Notifications
 - Connection
 - Custom Settings
 - Device Advanced
 - Device Info
 - Proxy
 - Security Settings
 - User Registration
7. Click **OK**.

Managing Properties of Application Connection Template

Manage the different categories of properties set for application connection templates.

1. In the left navigation pane, click the **Applications** node.
2. In the right administration pane, click the **Application Connection Templates** tab.
3. Select the application connection template from the list and click **Properties**.
4. In the Template dialog, select the category you want to edit and modify the property and value.
5. Click **OK**.

Deleting an Application Connection Template

Delete an application connection template.

1. In the left navigation pane, click the **Applications** node.
2. In the right administration pane, click the **Application Connection Templates** tab.
3. Select the application connection template from the list and click **Delete**.

4. Click **OK** on the confirmation dialog.

Application Connection Properties

Application connection properties are used to create application connections and application connection templates.

Apple Push Notification Properties

Apple push notification properties allow iOS users to install client software on their devices. This process requires you to create a different e-mail activation message using the appropriate push notification properties.

- **APNS Device Token** – the Apple push notification service token. An application must register with Apple push notification service for the iOS to receive remote notifications sent by the application’s provider. After the device is registered for push properly, this should contain a valid device token. See the iOS developer documentation.
- **Alert Message** – the message that appears on the client device when alerts are enabled. Default: `New items available`.
- **Delivery Threshold** – the frequency, in minutes, with which groupware notifications are sent to the device. Valid values: 0 – 65535. Default: 1.
- **Sounds** – indicates if a sound is made when a notification is received. The sound files must reside in the main bundle of the client application. Because custom alert sounds are played by the iOS system-sound facility, they must be in one of the supported audio data formats. See the iOS developer documentation.

Acceptable values: true and false.

Default: true

- **Badges** – the badge of the application icon.

Acceptable values: true and false

Default: true

- **Alerts** – the iOS standard alert. Acceptable values: true and false. Default: true.
- **Enabled** – indicates if push notification using APNs is enabled or not.

Acceptable values: true and false.

Default: true

Application Settings

Application settings display details that identify the Application Identifier, Domain, Security Configuration of an application connection template

- **Automatic Registration Enabled** – the value is set to **True** when the application connection registration is carried out automatically.
- **Application Identifier** – the application identifier registered on SCC.

- **Domain** – the domain selected for the connection template.
- **Security Configuration** – the security configuration defined for the connection template.

BlackBerry Push Notification Properties

BlackBerry push notification properties allow BlackBerry users to install messaging client software on their devices.

Property	Description
Enabled	Enables notifications to the device if the device is offline. This feature sends a push notification over an IP connection only long enough to complete the Send/Receive data exchange. BlackBerry Push notifications overcome issues with always-on connectivity and battery life consumption over wireless networks. Acceptable values: true (enabled) and false (disabled). If this setting is false, all other related settings are ignored. Default: true
Delivery threshold	The minimum amount of time the server waits to perform a push notification to the device since the previous push notification (in minutes). This controls the maximum number of push notifications sent in a given time period. For example, if three push notifications arrive 10 seconds apart, the server does not send three different push notifications to the device. Instead they are sent as a batch with no more than one push notification per X minutes (where X is the delivery threshold). Acceptable values: 0 – 65535. Default: 1
Push listener port	The push listener port reported by the device on which it listens for notifications. This port is automatically assigned by the client. For example, if there is another application already listening on this port, a free port is searched for. Default: 5011
Device PIN	Every Blackberry device has a unique permanent PIN. During initial connection and settings exchange, the device sends this information to the server. Unwired Server uses this PIN to address the device when sending notifications, by sending messages through the BES/MDS using an address such as: Device="Device PIN" + Port="Push Listener port". Default: 0
BES Notification Name	The BES server to which this device's notifications are sent. In cases where there are multiple BES servers in an organization, define all BES servers.

Connection Properties

Connection properties define the connection information used by Unwired Server to relate a user to a device.

- **Activation Code** – the original code sent to the user in the activation e-mail. Can contain only letters A – Z (uppercase or lowercase), numbers 0 – 9, or a combination of both. Acceptable range: 1 to 10 characters.

- **Farm ID** – a string associated with the Relay Server farm ID. Can contain only letters A – Z (uppercase or lowercase), numbers 0 – 9, or a combination of both. Default: 0.
- **Server Name** – the DNS name or IP address of the Unwired Server, such as "myserver.mycompany.com". If using Relay Server, the server name is the IP address or fully qualified name of the Relay Server host.
- **Server Port** – the port used for messaging connections between the device and Unwired Server. If using Relay Server, this is the Relay Server port. Default: 5001.
- **Synchronization Server Host** – the server host name used for synchronization.
- **Synchronization Server Port** – the port used for synchronization.
- **Synchronization Server Protocol** – the synchronization protocol - HTTP or HTTPS.
- **Synchronization Server Stream Parameters** – the server stream parameters.
- **Synchronization Server URL Suffix** – the server URL suffix.

Note: Some settings are visible but not in use by client (replication native application) at this time.

Custom Settings

Define one of four available custom strings that are retained during reregistration and cloning.

Change the property name and value according to the custom setting you require. The custom settings can be of variable length, with no practical limit imposed on the values. You can use these properties to either manually control or automate how workflow-related messages are processed:

- **Manual control** – an administrator can store an employee title in one of the custom fields. This allows employees of a specific title to respond to a particular message.
- **Automated** – a developer stores the primary key of a back-end database using a custom setting. This key allows the database to process messages based on messaging device ID.

Device Advanced Properties

Advanced properties set specific behavior for messaging devices.

- **Relay Server URL Prefix** – the URL prefix to be used when the device client is connecting through Relay Server. The prefix you set depends on whether Relay Server is installed on IIS or Apache. Acceptable values:
 - For IIS – use `/ias_relay_server/client/rs_client.dll`.
 - For Apache – use `/cli/iasrelayserver`.
- **Allow Roaming** – the device is allowed to connect to server while roaming. Acceptable values: true and false. Default: true.
- **Debug Trace Size** – the size of the trace log on the device (in KB). Acceptable values: 50 to 10,000. Default: 50.
- **Debug Trace Level** – the amount of detail to record to the device log. Acceptable values: 1 to 5, where 5 has the most level of detail and 1 the least. Default: 1.

Administer

- **Device Log Items** – the number of items persisted in the device status log. Acceptable values: 5 to 100. Default: 50.
- **Keep Alive (sec)** – the Keep Alive frequency used to maintain the wireless connection, in seconds. Acceptable values: 30 to 1800. Default: 240.

Device Info Properties

Information properties display details that identify the mobile device, including International Mobile Subscriber identity (IMSI), phone number, device subtype, and device model.

- **IMSI** – the International Mobile Subscriber identity, which is a unique number associated with all Global System for Mobile communication (GSM) and Universal Mobile Telecommunications System (UMTS) network mobile phone users. To locate the IMSI, check the value on the SIM inside the phone.
- **Phone Number** – the phone number associated with the registered mobile device.
- **Device Subtype** – the device subtype of the messaging device. For example, if the device model is a BlackBerry, the subtype is the form factor (for example, BlackBerry Bold).
- **Model** – the manufacturer of the registered mobile device.

Proxy Properties

(Applies only to Online Data Proxy) Proxy properties display details that identify the application and push endpoints.

- **Application Endpoint** – the back-end URL where the application service document is available
- **Push Endpoint** – the server URL where all the notifications are forwarded to.

Security Settings

Security settings display the device security configuration.

- **E2E Encryption Enabled** – indicate whether end-to-end encryption is enabled or not: true indicates encryption is enabled; false indicates encryption is disabled.
- **E2E Encryption Type** – use RSA as the asymmetric cipher used for key exchange for end-to-end encryption.
- **TLS Type** – use RSA as the TLS type for device to Unwired Server communication.

Note: These settings are visible, but not in use by client (replication native application) at this time.

User Registration

User registration specifies details of the activation code that is sent to a user to manually activate an application on the device.

- **Activation Code Expiration (Hours)** – indicates how long an activation code is valid. The default is 72 hours.

- Activation Code Length – indicates the length of the activation code, as in number of alphanumeric characters. The default is 3.

Administer

Deploy

Deployment is a routine administration task that manages the life cycle of a mobile business object (MBO) package on the Unwired Server. Deployment makes a package available to the runtime environment, so that it can be administered or accessed by client devices. Deployment is similar to, but not the same as, exporting and importing packages between multiple cluster environments.

Unwired Platform supports the development and subsequent deployment of:

- Package archives
- Mobile Workflow archives

Depending on the package type, the deployment steps can vary.

Packages

Packages are collections of MBOs that are related by application use and authorship and grouped according to maintenance or distribution. Packages are initially created by developers, but are deployed and maintained on a production Unwired Server by administrators.

Administrators cannot change the name of a package if one has been defined by the development team. You can, however, create new package versions when you make an upwardly incompatible change to an existing application. In this case, leave both versions of the package running until every one of the remote client applications has been upgraded to the latest version; only then should you delete the old package version.

Although each mobile business object (MBO) type has unique properties and data sources, MBOs within a package used by an application may be of different types.

Note: You must deploy a package before you can configure or manage it. Package administration tasks vary, depending on the type of package you deploy.

Deploying Packages

Use the Deploy wizard to make packages available on the Unwired Server.

Prerequisites

If your developers have created a custom filter for the mobile business object you are deploying, you must copy class files to the primary server before you deploy the package that uses those filters. If you copy the filters to a slave server by mistake, they are deleted when you deploy the package to the primary server.

Deploy

To locate the name of the master, look at the server list in Sybase Control Center. If a particular server is the master server of a cluster, it will be labeled as "primary" in the left navigation pane.

Task

Deploying is the process whereby whole or part of a mobile application is loaded onto an Unwired Server as one or more deployment units. Unwired Server can then make these units accessible to users via a client application that is installed on a mobile device.

Because the deployment unit file contains package name and other information, you do not need to select a package from the list of available packages; Unwired Server creates the package automatically according to what has been defined in the deployment file.

Note: If the connection properties of an MBO or operation use credentials that have been customized manually by a developer, the back-end data sources connection properties of these MBOs and operations cannot be updated by an administrator when using the Deploy Wizard.

Instead, the administrator can update these properties after the MBO or operation is deployed to an Unwired Server. When the server connection is created on Unwired Server, the administrator can then change connection properties in Sybase Control Center by clicking on the Connections node in the left navigation pane.

Launching the Deploy Wizard

Launch the Deploy wizard when the packages you require have already been created, or when you want Unwired Server to create a package for you from a deployment unit file.

1. In the left navigation pane, expand the **Domains** folder.
2. Choose a domain name, then select **Packages**.
3. In the right administration pane, click the **General** tab.
4. Click **Deploy**.

Follow the instructions in the wizard to configure a package so it can be deployed.

Configuring Deployment Properties

Set the properties for packages being deployed on Unwired Server.

A deployment file and an optional deployment descriptor file are created by developers in Unwired Workspace. These files are typically delivered for deployment to a production version of the Unwired Server by an administrator.

Note: If the package is deployed to the primary server, it is cluster-level operation.

1. When the **Deploy** wizard loads, click **Next**.
2. Review the **Deployment File** name, or click **Browse** to navigate to the appropriate file.
You can select either a single deployment unit (.XML) or an archive file (.JAR).

The name of the **Package Name** appears. If this package does not already exist, the wizard displays a message that indicates the new package will be created. The name cannot exceed 64 characters or include any periods (".").

3. Select a **Deployment Mode**.

The deployment mode determines how the deployment process handles the objects in a deployment unit and package. Which value you choose depends on whether or not a package of the same name already exists on Unwired Server. Allowed values are:

- **UPDATE** – updates the target package with updated objects. After deployment, objects in the server's package with the same name as those being deployed are updated. By default, deploymentMode is UPDATE.
- **NOCLOBBER** – deploys the package only if there are no objects in the target server's package that have the same name as any of those objects being deployed.
- **REPLACE** – replaces any of the target objects with those in the package. After deployment, the servers package contains only those objects being deployed.
- **VERIFY** – do not deploy package. Only return errors, if any. Used to determine the results of the UPDATE deploy mode.

If the deployment mode is specified both in the descriptor file and the command-line, the command-line deploymentMode option override the deployment mode specified in the descriptor file.

4. If you did not choose a deployment archive as your deployment file, you may browse and select an optional deployment **Descriptor File**:

5. Click **Next**.

6. Select a **Domain** to deploy the package to.

7. Select a **Security Configuration** for the package.

8. Click **Next**.

The Configure Role Mapping page appears.

Deployment Archives

An archive is produced after a developer creates a package profile and executes a build on a package. This archive can only be created in the Eclipse edition of Unwired Workspace.

In Unwired Workspace, a developer executes a build process so that it creates a . jar archive file, which contains both a deployment unit and a corresponding descriptor file. A deployment archive can be delivered to an administrator for deployment to a production version of the Unwired Server.

Deployment Descriptors

A deployment descriptor is an XML file that captures changes to the deployment unit during deployment. Those changes are then used when a package is redeployed.

A deployment descriptor is not required to deploy the deployment unit.

Deploy

A deployment descriptor is created either after a developer creates a package profile and executes a build on a package, or when the developer deploys a package to a development edition server from Unwired WorkSpace. The file contains this information:

- The deployment mode
- The target package that descriptor applies to
- The endpoint information that overrides specific endpoints defined for MBOs or operations in the deployment unit
- The domain and named security that the package applied to
- Role mappings

This information is specific to each deployment unit; therefore, you cannot apply a descriptor from another package to a deployment unit.

Setting the Mapping State

Map roles for a package by setting the mapping state. Mapping behavior is determined by the state that exists for the logical role. You can select **AUTO** or **NONE**; a third state, **MAPPED**, is set automatically after you manually map a physical role to the selected logical role.

You can set the mapping state either when managing roles, or earlier, during package deployment. If your logical roles for a package do not automatically match the role names registered in the back-end security system, map corresponding logical and physical names to ensure that users can be authorized correctly.

1. For package-specific role mapping, select and deploy an available package. Follow the wizard prompts until you reach the **Configure Role Mapping** page for the target package.
2. Change the mapping for a logical role, if required:
 - To change the state to either **NONE** or **AUTO**, click the list adjacent to the logical role and click the appropriate option.
 - To change the role mapping itself, click the drop-down list adjacent to the logical role and choose **Map Role**. This command displays the **Role Mappings** dialog that allows you to manually set the physical role mappings. The **Role Mappings** dialog displays the name of the logical role you are mapping in the text area of the dialog. Once saved, the state automatically changes to **MAPPED**.
3. Click **Next**.
The **Server Connection** page appears.

Deployment-time role mapping is done at the package level. Once the package is deployed, you can change the role mapping by going to the **Role Mapping** tab for the desired package. You can also set the role mapping for each security configuration at the domain level. This allows the role mapping to be shared across packages for the common logical roles. Changing role mapping at the domain level will result in role mapping changes in other domains where the same security configuration is referenced.

Updating Server Connection Properties

Configure the production version of server connection properties. Typically, the endpoint and role mapping a developer would use are not the same for a production system. Administrators must reset these properties accordingly.

This step allows you to change connection profiles used for development (design time) to an appropriate server-side connection. For example, your development environment might permit access to certain systems that the Unwired Server prohibits.

Note: If the connection properties of an MBO or operation use credentials that have been customized manually by a developer, the back-end data sources connection properties of these MBOs and operations cannot be updated by an administrator when using the Deploy Wizard.

Instead, the administrator can update these properties after the MBO or operation is deployed to an Unwired Server. When the server connection is created on Unwired Server, the administrator can then change connection properties in Sybase Control Center by clicking on the Connections node in the left navigation pane.

1. Review the connection properties.
2. Depending on how the properties are configured, choose an appropriate option:
 - If the properties are configured by the developer as a set of connection properties, you can edit properties as needed. To edit a property, click a field in the **Value** column and change the value as required.
 - If the properties are configured as an endpoint, choose the connection pool you want Unwired Server to use. You cannot alter any of these properties.
3. If you want the changes to apply to operations only, click the corresponding check box at the bottom of the table.
4. Click **Next**.

The connection properties are updated. The changes you make are displayed in the Summary page in the Endpoint Updates section.

Reviewing the Deployment Summary

Review the properties you have supplied before deploying the package to Unwired Server. This allows you to change errors before making the deployment units available via the package created for that purpose.

1. Review all three sections of the deployment summary. If anything is incorrect, click **Back** and correct errors.
2. To create a deployment descriptor:
 - a) Click **Create deployment descriptor**.
 - b) Browse to the location you want to save the file and change the default name if required.

Deploy

c) Click **OK**.

You can use the deployment descriptor to redeploy the deployment unit without having to repeat these steps in the deployment wizard.

3. Click **Finish** to deploy the file and create the package on Unwired Server.

Deploying to the server may take some time to complete. However, a status message appears above the **General** tab indicating the success or failure of the attempt.

Package Management

There are common package management activities used by all package types: replication, messaging, unified, DOE-C, or ODATA.

These activities include:

Importing Package Contents

Import package contents after they have been exported from another Unwired Server.

Note: A package can only be imported in the same domain as it was exported. Therefore it is required that same domain exists in the server before the import. Also, it is required to create domains in the same order in both export and import server environment to ensure that an internal ID assigned to the domain in both environment matches. You can verify the internal ID assigned to a domain by looking at the prefix used in the package folder in the exported zip.

1. In the left navigation pane, expand the **Packages** folder.
2. In the right administration pane, click **Import**.
3. Enter a local file to import, or click **Browse** to navigate to the file.
4. Click **OK**.

Exporting Packages

Export a package to bundle one or more MBOs and package options in the selected package to create a new instance of a deployment archive. Typically, export is used to move the package contents onto another Unwired Server, however you can only import packages to the same domain as it was exported from.

Note: It is a requirement to create domains in the same order in both the export and import server environment to ensure that an internal ID assigned to the domain in both environment matches. You can verify the internal ID assigned to a domain by looking at the prefix used in the package folder in the exported zip.

1. In the left navigation pane, expand the **Packages** folder.
2. In the right administration pane, check the box adjacent to the name of the package and click **Export**.
The **Export Package** dialog appears.

3. Check the appropriate export package options from the list: **Include synchronization tracing**, **Include package logging**, **Include role mappings**, and **Include device notification templates** (for RBS packages only).

These selections determine which package settings are retained for the new instance of the package. The current security configuration for the package is automatically applied to the exported package.

4. Click **Next**.
5. Select the file system target for the exported contents and click **OK**.

Note: Ensure that you do not hide file type extension when you name the export package; otherwise, when the *.zip extension becomes invisible, which adversely affects the outcome of the export process.

A status message on the General tab indicates the success or failure of the export transaction. If successful a ZIP file is created in the location you specified. You can then use this file to import the package.

Next

Deliver the file to the appropriate person or deploy the exported package on the appropriate server.

Deleting a Package

Delete a package when you want to permanently remove all elements deployed on an Unwired Server. If you do not want to permanently prohibit access by removing these files, consider disabling the package instead.

1. In the left pane, click the Unwired Server you are currently logged in to.
This expands the list of Unwired Platform components you can manage, provided that you have the correct permissions to do so.
2. Click **Domains > DomainName > Packages**.
3. To delete a package from the this list, select one or more packages and click **Delete**.
A confirm dialog box appears.
4. Click **OK** to confirm the deletion.

The package is removed from Unwired Server.

Enabling and Disabling a Package

Enable or disable a package to allow or prohibit device access to the package. Disabled packages are still available to Sybase Control Center for Unwired Server. By default, all packages are enabled.

When you disable a package, the server unloads all of its elements from memory. Disabling a package prevents the Unwired Server from loading that package at start-up.

Note: You cannot disable a SAP Data Orchestration Engine Connector (DOE-C) package.

1. In the left navigation pane, click **Packages**.
2. In the right administration pane, select the **General** tab.
3. Select the box adjacent to the package you want to enable or disable.
You can select more than one package to apply the same change to multiple components.
4. Depending on the current status of the package, perform one of:
 - Enable – if the package listed shows a status of disabled, click **Enable**.
 - Disable – if the package listed shows a status of enabled, click **Disable**. The package remains disabled until you or another administrator enables it; restarting Unwired Server does not enable the package.

Configuring Synchronization Groups

Configure a synchronization group when a collection of MBOs must be synchronized at the same time. The synchronization group defines the logical unit of synchronization and data notifications required (depending on the payload protocol), as well as the synchronization frequency.

Determine the frequency with data changes are generated by Unwired Server for synchronization of mobile business objects (MBOs). The delivery time is determined by this equation: $\text{delivery of data change} = \text{schedule repeat value} + \text{change detection interval value} + \text{device push settings}$. There may be an additional 10 second delay, because the internal thread for performing the change detection runs every 10 seconds.

1. In the left navigation pane, expand the **Packages** folder and select the package containing the sync group you want to configure.
2. Select the desired sync group and click **Properties**.
3. Select a change detection interval. This value determines how frequently Unwired Server looks for changes to MBOs, and generates push notifications. The default is 1 hour.
4. Click **OK**.

Configuring a Cache Group

Select and configure a cache group. The cache is part of the Unwired Server cache database (CDB) that is used to store data that is uploaded and downloaded from EIS servers and mobile clients during synchronization.

Cache group configuration differs depending on whether the cache group is defined as "on demand" or "scheduled" during development.

MBO Data in the CDB

A data cache is a copy of MBO data that is stored in a specific area of the cache database (CDB). It is used as the data repository for replication and messaging MBOs that are deployed

to Unwired Server. "CDB" and "cache" and "MBO data" can sometimes be used interchangeably, even though the CDB includes runtime data as well.

When cache data is updated (either with an on-demand or scheduled cache refresh), the remote client database eventually retrieves the updated data from the server's copy of MBO data in the CDB by synchronization.

By giving applications a normalized and uniform view of corporate data, organizations can:

- Lower the barrier to data behind corporate firewalls
- Support development of mobile applications that interact with multiple enterprise back-ends
- Reduce back-end load caused by device client requests

Configuring On Demand Cache Group Properties

Specify the duration of cache data validity by configuring Unwired Server updates to mobile business object (MBO) data for an on demand cache group.

Note: The developer configures a cache group as either on demand or scheduled. If the cache group is "scheduled," the Cache tab is not configurable in the Cache Group Properties dialog.

1. In the left navigation pane, expand the **Packages** folder, and select the package for which you want to configure cache settings.
2. In the right administration pane, click the **Cache Group** tab.
3. Select the cache group you want to configure and click **Properties**.
4. In the **Cache Properties** dialog, enter an expiry for the **Cache Interval** in seconds, minutes, or hours.

The cache interval determines how frequently Unwired Server updates the cache database with changes to enterprise data. See *On Demand Cache Refreshes* in the list of links below.

5. Click **OK**.

On Demand Cache Refreshes

Cache groups designated as "on demand" during development use cache intervals to balance how frequently the object updates enterprise data with the amount of network traffic required to maintain that data.

Unwired Server keeps a local copy of enterprise data in the cache database (CDB), and uses an intricate mechanism to manage updates between the CDB and the EIS servers. When data is updated, the remote client database eventually gets updated data from this local copy in the CDB. The caching mechanism allows MBOs to retrieve updated data even if back-end servers fail.

You must choose an appropriate cache interval for your system, since this value determines how frequently the CDB is updated with data from the EIS. The cache interval must be configured according to business needs. A higher value for the cache may retain stale data, however, a lower value increases the backend EIS load and may impede the client application's

performance, because Unwired Server queries the back-end information servers more frequently to look for changes and possibly update the CDB copy.

Frequent queries typically put a higher load on the servers, as well as use more network bandwidth on the server side. While the cache interval does not affect the bandwidth required between the synchronization server and device client applications, nor the performance characteristics of the client applications, the interval you choose can delay synchronization if Unwired Server must first update many records in the CDB.

For example, if the cache interval is 0, each time a client application synchronizes, there is a pause while the Unwired Server rereads data from the EIS and updates the CDB. If, however, the cache interval is greater than 0, then the wait time depends on how long ago the data was refreshed. If the synchronization falls within a recent cache update, synchronization is almost immediate.

Configuring Scheduled Cache Group Properties

Specify the duration of cache data validity by configuring Unwired Server updates to mobile business object (MBO) data for a scheduled cache group.

Prerequisites

You can configure a schedule refresh for a cache only if the developer enables the cache group as "scheduled" during development. Otherwise, the Schedule tab is not configurable in the Cache Group Properties dialog.

Task

1. In the left navigation pane, expand the contents of the **Packages** folder and select the package for which you want to display properties.
2. In the right administration pane, click the **Cache Group** tab.
3. Select the cache group you want to configure and click **Properties**.
4. In the **Schedule** tab of the Cache Properties dialog, set the frequency of the refresh by selecting an appropriate **Schedule Repeat**: hourly, daily or custom.

This property determines what other schedule properties you must configure. Each option is documented in a separate topic which further discusses the details for each frequency type. For more details, see the corresponding topic.

Scheduling an Hourly or Daily Refresh

Scheduling an hourly or daily cache refresh means that information is fetched from the enterprise information server (EIS) and populated into the cache on either of these hourly or daily frequencies according to the schedule and range of time you configure.

The Schedule tab in the Cache Property dialog displays options appropriate for configuring this type of schedule.

1. Select either **Hourly** or **Daily** as the **Schedule Repeat** criteria.
2. (Optional) If you want to set a range to control which days the schedule refresh runs, configure a start date and time, an end date and time, or day of week (if applicable).
 - Select **Start Date** to set a date for which the first execution of the scheduled refresh is performed. To be more specific, you can also select **Start Time** to specify a start time. In this case, the refresh cannot begin until a given time on a given day has been reached. A start date and time are inclusive.
If you do not set a start date and time, then, by default, the date and time that Unwired Server starts is used.
 - Select **End Date** to set a date that ends the repeating refresh transactions for a package. To be more specific, you can also select **End Time** to specify an end time. An end date and time are exclusive. This means that a refresh transaction runs up to, but does not include, the end time. For example, if a schedule has a start time of 13:00 and an end time of 16:00 and repeats every hour, it runs at 13:00, 14:00, and 15:00, but not at 16:00.
If you do not set an end date and time, then, by default, the date and time that Unwired Server stops is used.
 - Select **Specify Week Days** to select the days of the week that the refresh transaction runs. This means that for the days you select, the refresh runs every week on the day or days you specify. A weekday is inclusive. This means that any day you choose is included in the frequency. All others are excluded.

When the schedule expires, the automatic refresh you configured terminates, unless the end user initiates a refresh.

3. Click **Save**.

Scheduling a Custom Refresh

Scheduling a custom cache refresh is the most flexible of all cache refresh schedules. This means that information is fetched from the enterprise information system (EIS) according to the schedule repeat interval you specify.

The Schedule tab in the Cache Property window displays options appropriate for configuring this type of schedule.

1. Select **Custom** as the schedule repeat criteria.
2. Specify a repeat **Interval**, in minutes or seconds, to determine how often the cache refresh occurs.
This interval determines how frequently Unwired Server updates the cache database with changes to enterprise data. The default is 0 seconds, which means the mobile business object retrieves the data from the enterprise information server (EIS) on every playback request. If you choose something other than 0 seconds, the data is held by the cache for the duration of the specified interval.
3. (Optional) To set a range to control which days the schedule refresh runs, configure a start date and time, end date and time, or day of week (if applicable).

- Select **Start Date** to set a date for which the first execution of the scheduled refresh is performed. To be more specific, you can also select **Start Time** to specify a start time. In this case, the refresh cannot begin until a given time on a given day has been reached. A start date and time are inclusive.
If you do not set a start date and time, then, by default, the date and time that Unwired Server starts is used.
- Select **End Date** to set a date that ends the repeating refresh transactions for a package. To be more specific, you can also select **End Time** to specify an end time. An end date and time are exclusive. This means that a refresh transaction runs up to, but does not include, the end time. For example, if a schedule has a start time of 13:00 and an end time of 16:00 and repeats every hour, it runs at 13:00, 14:00, and 15:00, but not at 16:00.
If you do not set an end date and time, then, by default, the date and time that Unwired Server stops is used.
- Select **Specify Week Days** to select the days of the week that the refresh transaction runs. This means that for the days you select, the refresh runs every week on the day or days you specify. A weekday is inclusive. This means that any day you choose is included in the frequency. All others are excluded.

When the schedule expires, the automatic refresh you configured terminates, unless the end user initiates a refresh.

4. Click **Save**.

Scheduled Cache Refreshes

A schedule-driven cache refresh is a background task that runs between a configured start and endpoint at scheduled intervals during normal server operation.

A schedule-driven cache refresh defines a contract between Unwired Server and back-end information servers. Normally, data is retrieved from a server (for example, a database, and an SAP repository, or a Web service) when a device user synchronizes. If the administrator wants the data to be preloaded, he or she configures the Unwired Server repeat interval to expedite data updates on the device.

Two properties configure the cache refresh schedule, which is used with a subscription to synchronize data for mobile business objects (MBOs).

- **Schedule repeat** – determines the time frame when data is refreshed. If you set up a schedule to repeatedly refresh data, information is always refreshed. Set the schedule to meet business application requirements for data consistency.
As an administrator, you may also use a schedule repeat to look for data changes and alert subscribed clients to synchronize when there are changes. Keep in mind, however, that the actual detection of changes and sending of data (for messaging payloads) or notifications (for replication payloads) depends on the:
 - **Change detection interval** property of the synchronization group for the package.
 - **Notification threshold** property of subscriptions for replication payloads.

- Push related device settings for messaging payloads.
- Repeat interval – determines how often Unwired Server updates the cache with changes to backend data.

Online Refresh Policy

MBOs that use an Online refresh policy indicates that the MBOs are to be used only in Workflow applications where access to real-time enterprise information system (EIS) data is required (cache validity is zero).

Data is valid in the Unwired Server cache only until delivery and immediately invalid. You cannot modify the cache or schedule of the Online policy. Expired data is purged from the cache based on a schedule at the domain level.

DCN Refresh Policy

The cache refresh and schedule options are disabled for DCN (data change notification) policy, since data never expires and is not refreshed based on client demand or a schedule.

Cache data does not expire until a cache invalidate operation is invoked or a data change notification request is received from the enterprise information system (EIS).

Purging a Cache Group

Physically delete data that has been logically deleted from the cache. Cached data is marked as logically deleted when certain activities occur in the client application or back end.

1. In the left navigation pane of Sybase Control Center, expand the **Packages** folder and select the package to configure.
2. In the right administration pane, select the **Cache Group** tab.
3. Click **OK**.

Purging the Synchronization Cache Manually

You can manually purge the synchronization at the package level. Mobile business objects (MBOs) contained in a cache group using the online policy are deleted.

1. In the left navigation pane, expand the **Domains** folder and select a domain.
2. Navigate to the **Packages** folder, and select a package.
3. In the right pane, select the **Cache Group** tab.
4. On the **Cache Group** tab, click **Purge**.
5. Click **OK** to purge immediately based on your selections.

Assigning Package-Level Security

Assign security configurations at the package level to override those set at the domain-level.

Selecting a Security Configuration for a Package

Designate a security configuration for a package in Sybase Control Center. This is a required step during package deployment, but you can later change the security configuration.

The administrator must create a security configuration in the cluster and assign it to the domain where the package is deployed before the deployer can assign the security configuration to the package.

1. In the left navigation pane, expand the **Packages** folder, and select the package to configure.
2. In the right administration pane, click the **Settings** tab.
3. Select a security configuration.

The security profiles that appear in this list have been created by a platform administrator and assigned to the domain.

4. Click **Save**.

Mapping Roles

Configure role mapping to authorize client requests to access MBOs and operations. For each security configuration, platform and domain administrators can manage logical role mappings at the package level or at a domain level. Use the corresponding domain or package node in the left navigation pane to configure role mappings accordingly.

Set an appropriate mapping state for each logical role. The state you choose allows you to disable logical roles, allow logical roles to be automatically mapped, or manually define which logical roles are mapped to one or more physical roles. The states of AUTO or NONE require the least administration.

If a developer has defined a logical role, mapping is not required; the logical role is matched to the physical role of the same name and is therefore automatically mapped.

Note: Changes to domain-level role mapping are applied to all domains that share the same security configuration. Likewise, changes to package-level role mapping apply to all instances of the affected package that use the same security configuration, even if the package is deployed in multiple domains.

Setting the Mapping State

Map roles for a package by setting the mapping state. Mapping behavior is determined by the state that exists for the logical role. You can select AUTO or NONE; a third state, MAPPED, is set automatically after you manually map a physical role to the selected logical role.

You can set the mapping state either when managing roles, or earlier, during package deployment. If your logical roles for a package do not automatically match the role names registered in the back-end security system, map corresponding logical and physical names to ensure that users can be authorized correctly.

1. For package-specific role mapping, select and deploy an available package. Follow the wizard prompts until you reach the Configure Role Mapping page for the target package.
2. Change the mapping for a logical role, if required:
 - To change the state to either NONE or AUTO, click the list adjacent to the logical role and click the appropriate option.
 - To change the role mapping itself, click the drop-down list adjacent to the logical role and choose **Map Role**. This command displays the Role Mappings dialog that allows you to manually set the physical role mappings. The Role Mappings dialog displays the name of the logical role you are mapping in the text area of the dialog. Once saved, the state automatically changes to MAPPED.
3. Click **Next**.
The Server Connection page appears.

Deployment-time role mapping is done at the package level. Once the package is deployed, you can change the role mapping by going to the Role Mapping tab for the desired package. You can also set the role mapping for each security configuration at the domain level. This allows the role mapping to be shared across packages for the common logical roles. Changing role mapping at the domain level will result in role mapping changes in other domains where the same security configuration is referenced.

Mapping a Physical Role Manually

Use the Role Mappings dialog to manually map required physical roles for a logical role when physical and logical role names do not match. If names do not match, the AUTO mapping state does not work.

Prerequisites

Unwired Platform cannot query all supported enterprise security servers directly; for successful authentication, you must know the physical roles your back-end systems require.

Task

You can map a logical role to one or more physical roles. You can also map multiple logical roles to the same physical role. If a role does not exist, you can also add or delete names as needed.

1. Review the list of existing physical role names that you can map to the logical role you have selected.
2. If a role that you require does not appear, enter the **Role name** and click the + button. The role name appears in the **Available roles** list with an asterisk (*). This asterisk indicates that an available role was added by an administrator, not a developer.
3. To remove a role you no longer require from the **Available roles** list, select the name and click the **x** button adjacent to the **Role name** field.
The role is removed and can no longer be mapped to a logical role.

Deploy

4. To map a logical role that appears in the text area of the Role Mappings dialog to a physical role:
 - a) Select one or more **Available roles**.
 - b) Click **Add**.
5. To unmap a role:
 - a) Select one or more **Mapped roles**.
 - b) Click **Remove**.
The roles are returned to the **Available roles** list.
6. Click **OK** to save these changes.

Once a logical role has been manually mapped, the mapping state changes to **MAPPED**. The roles you have mapped appear in the active Physical Roles cell for either a package-specific or server-wide role mappings table.

Mapping State Reference

The mapping state determines the authorization behavior for a logical name instance.

State	Description
AUTO	Map the logical role to a physical role of the same name. The logical role and the physical role must match, otherwise, authorization fails.
NONE	Disable the logical role, which means that the logical role is not authorized. This mapping state prohibits anyone from accessing the resource (MBO or Operation). Carefully consider potential consequences before using this option.
MAPPED	A state that is applied after you have actively mapped the logical role to one or more physical roles. Click the cell adjacent to the logical role name and scroll to the bottom of the list to see the list of mapped physical roles.

Logical Roles

Most logical roles are defined by MBO modelers or.

MBO modelers define logical roles in their models, either at the MBO or operation level:

- When logical roles are assigned to an MBO, that defines the role based access control (RBAC) policy for who can read data from that MBO.
- When logical roles are assigned to an MBO operation, it defines the policy for who can execute that operation.

After the MBO is deployed to a production Unwired Server, an Unwired Platform administrator may create the logical and physical role mappings. This usually occurs after consulting both:

- The MBO application developer to understand which logical roles are meant to protect access.
- The backend security administrator to understand which physical roles are available.

After such consultation, the platform administrator may deem it appropriate to create new backend roles and assign subjects to them to get appropriate groupings.

In the absence of explicit mapping, the default role mapping is set to AUTO, which is equivalent of logical role mapping to a physical role of the same name, in the underlying provider of that security configuration.

SUP DCN User Role

The SUP DCN User is a logical role that Unwired Platform uses to authorize any DCN event: updating data in the cache, executing an operation, or triggering a workflow package.

Before any DCN event is submitted, the person or group mapped to this role must be authenticated and authorized by the security configuration. By default, SUP DCN User is automatically available to all new security configurations. However, the underlying default varies depending on the environment in use.

Before this logical role can be used, SUP DCN User must be mapped to a physical role in the enterprise security repository, and the user who performs DCN must be in that physical role.

- To map the SUP DCN User to a user in the underlying security repository, the user name must be first defined in Sybase Control Center as a physical role that is mappable. Then, SUP DCN User role can be mapped to a physical user or to a physical role from Sybase Control Center. For example, to map SUP DCN User to a user that is not in the security repository, use the format `user:User`.

If you are supporting multiple domains, the user name must also include the named security configuration for the package the DCN is targeted for, by appending `@DomainSecurityConfigName` as a suffix to that name. Suppose you have two packages (PKG_A, PKG_B) deployed to two domains (Domain_A, Domain_B) respectively. Further, assume that PKG_A in Domain_A has been assigned to the "admin" security configuration, whereas PKG_B in Domain_B has been assigned to the "alternateSecurityConfig" security configuration.

- A user doing DCN to PKG_A should identify him or herself as `User@admin`.
- A user doing DCN to PKG_B should identify him or herself as `User@alternateSecurityConfig`.

If you are using ActiveDirectory, and are using e-mail addresses for user names, definitions appear as `username@myaddress@DomainSecurityConfigName`.

The implementation varies, depending on the DCN service used:

- For workflows, because the resource the user is pushing data toward is a group of named users (users authenticated previously successfully against a certain security configuration), he or she must have the authorization to push to that particular security

Deploy

configuration. The user must be mapped to SUP DCN User in the security configuration for the workflow target.

- A user having SUP DCN User logical role in security configuration "mySecConfig1" must not have the right to push workflow DCN or regular DCN to a user or package associated with "mySecConfig2".

Physical Roles

Physical roles are named references to roles or groups that an administrator has defined on a back-end enterprise security provider. Mapping a logical role to a physical role allows authorization control in the Unwired Server. Replicate these names exactly, so that logical roles can be mapped correctly to the server role.

Physical roles are sourced from the enterprise security repositories. In some cases, the security providers configured by the administrator in Sybase Control Center are not able to reliably extract a list of physical roles from these repositories. In which case, Unwired Platform administrator must collaborate with the security administrator to get a list of physical roles. These must be manually entered in the role mapping stage when creating a security configuration in Sybase Control Center.

Viewing Asynchronous Operation Replays

View pending asynchronous operation replays for unified or replication packages already deployed to Unwired Server.

Prerequisites

Asynchronous operation replays are only available if the feature has been enabled as part of the cluster properties. See *Configuring General Cluster Properties*.

Task

1. In the left navigation pane, expand the **Packages** folder and select the deployed package.
2. In the right administration view, click **the Asynchronous Operation Replay** tab.
3. Review the requests displayed in the table for this package.

The number of replays are ordered in sequence along with:

- The user name that performed the operation.
- The name of the operation.
- The name of the MBO.

4. If the list is long, you can search for a particular entry:

- a) Select the field you want to search for: user, operation, or MBO name.
- b) Type the text string in the adjacent field.

This file allows the use of wildcards (that is, *). For example, if you selected a user name search, and you want to return replays for all users that start with A, you type A* as the search string.

Viewing and Changing Package Connection Properties

View and edit connection properties for packages.

1. In the left navigation pane, expand the **Packages** folder, select the package whose connection properties you want to view or change, and select the mobile business object.
2. In the right navigation pane, click the **Connection** tab.
3. If you wish to change any of the settings:
 1. In the left navigation pane, click the **Connections** icon.
 2. Select the checkbox for the **Connection Pool Name** that matches the name of the package whose connection properties you were just viewing.
 3. Click **Properties**.
 4. Make desired changes and click **Save**.

Note: For DOC-C connections, you must save the connection properties first, then test connection. For other connection types, you can test connection without saving.

5. Click **Test Connection**.

If the connection test is not successful, see *Connection Test Errors* in the *Troubleshooting* guide

Subscription Management

Manage subscriptions for packages.

Subscription management activities include:

Configuring Package Subscriptions

Configure subscriptions and subscription templates to allow the device user to be notified when information is available, depending on the subscription properties configured for a package. Subscription templates allow you to configure predefined properties for a synchronization group. A client's first synchronization for a specified synchronization group results in subscription creation using both the template and client-specified properties.

There are two main activities to set up a notification for a synchronization package.

Creating Replication Subscription Templates

Create a subscription template to specify synchronization targets and behavior for subscribed users. A template is useful to create a set of predefined values that are used frequently. Otherwise, a subscription is still automatically created for each client upon explicit indication of interest for a device notification.

This is an optional step; it is only required if an administrator wants to establish preset subscription properties. The subscription properties can be modified from device application, but only if the **Admin lock** property is disabled.

1. In the left navigation pane, expand the **Packages** folder and select the replication based sync package you want to configure.
2. In the right administration pane, click the **Subscriptions > Replication** tab.
3. From the menu bar, select **Templates**.
4. Click **New**.
5. In the New Template dialog, select settings for these options:
 - Synchronization Group – the group of MBOs that a client receives data change notifications for when data changes occur.
 - Notification Threshold – the length of time that must pass since a client's last synchronization before another notification is sent.
 - Admin Lock – (enable or disable) prevents device users from modifying the push synchronization state or sync interval value configured in the subscription. If the admin lock is disabled, the device client user can change these properties, and these changes take effect the next time the client user synchronizes the package to which the subscription applies.
 - Push – (enable or disable) if enabled, automatic server-initiated notifications are pushed to users when changes occur in the cache. If disabled, device users perform client-initiated synchronizations when they receive an outbound notification.

Note: If you intend to use push synchronization with BlackBerry devices, enable push synchronization in the BlackBerry server. See the BlackBerry server documentation for details.

6. Click **OK**.
The new subscription template appears in the list of templates.

Notifications are delivered to BlackBerry device clients using push-based notification settings and to Windows Mobile device clients using pull-based notification settings. The "poll every" setting determines the notification delivery behavior. See in Sybase Control Center online help.

Configuring Replication Subscription Properties

View and configure subscription properties for replication device users subscribed to deployed synchronization packages.

1. In the left navigation pane, expand the **Packages** folder and select the replication-based synchronization package you want to configure.
2. In the right navigation pane, click the **Subscriptions > Replication** tab.
3. From the menu bar, select **Devices**.
4. Check the box adjacent to a device user and click **Properties** to view these subscription properties:
 - Application Connection ID – the unique identifier for a user application connection.

- User – the name of the user associated with the application ID.
 - Package Name – the name of the package to which the subscription belongs.
 - Sync Counts – the total number of synchronizations for the subscription since the synchronization history was last cleared.
 - Notification Threshold – the length of time that must pass since a client's last synchronization before another notification is sent.
 - Last Sync Time – the date and time that the last synchronization for the subscription occurred.
 - Synchronization Group – the group of MBOs that a client receives data change notifications for when data changes occur.
 - Admin Lock – (enable or disable) prevents device users from modifying the push synchronization state or sync interval value configured in the subscription. If the admin lock is disabled, the device client user can change these properties, and these changes take effect the next time the client user synchronizes the package to which the subscription applies.
 - Push – (enable or disable) if enabled, automatic server-initiated notifications are pushed to users when changes occur in the cache. If disabled, device users perform client-initiated synchronizations when they receive an outbound notification.
5. Check the box adjacent to **Clear sync history** in order to erase stored synchronization details for the subscription.
 6. If you made changes to subscription properties, click **Save**. Otherwise, click **Cancel** to return to the Subscriptions tab.

Configuring Messaging Subscription Settings

View and edit user properties for messaging packages that allow you to manage messaging between Unwired Server and application users.

1. In the left navigation pane, expand the **Packages** folder, and select the package you want to configure.
2. In the right navigation pane, click **Subscriptions > Messaging** .
3. Check the box adjacent to an application connection and click **Device Settings**.
4. Configure these property categories, as required:
 - Connection
 - Custom Settings
 - Device Advanced
 - Device Info
 - User Registration
 - Apple Push Notifications (iPhone only)
 - BlackBerry Push Notifications
 - Proxy 'Security Settings

- Application Setting

Managing Deployed Package Subscriptions

Manage UNIFIED, messaging, and SAP Data Orchestration Engine connector (DOE-C) package subscriptions that specify the synchronization messages mobile device users receive.

Subscription management tasks include pinging, unsubscribing, recovering, suspending, resuming, resynchronizing, and logging subscriptions. Subscription tasks vary by the package type.

These subscription management tasks apply only to the package types specified in the table below. Perform each task in the Subscriptions tab of the deployed package you are managing.

Table 16. Subscription management tasks

Subscription task	Description	Summary	Package type
Ping	<p>Ensure that push information a user provides for a device is configured correctly.</p> <p>If the ping is successful, notifications and subsequent data synchronizations occur as defined by each subscription. If the ping fails, open the log and check for an incorrect host name or port number.</p>	<p>Select the box adjacent to the device ID, and click Ping.</p>	<p>UNIFIED (replication subscriptions)</p>
Unsubscribe	<p>Remove a subscription from Unwired Server.</p>	<p>Select the box adjacent to the device ID, and click Unsubscribe for UNIFIED packages, messaging packages, and DOE-C packages.</p> <p>For Windows Mobile, the device application must include the DatabaseClass.CleanAllData(); method for data to be unsubscribed correctly. If this method is not used, Unsubscribe and Subscribe could work unpredictably.</p>	<p>All</p>

Subscription task	Description	Summary	Package type
Recover	<p>Reestablish a relationship between the device and Unwired Server. Perform recovery under severe circumstances when a device is unable to successfully synchronize data.</p> <p>During subscription recovery, Unwired Server purges all enterprise data on the device. It retains the device ID and subscription information so that all data can then be resynchronized and loaded onto the device.</p>	Check the box adjacent to the subscription ID of the device, and click Recover .	Messaging
Suspend/resume	<p>Control the deactivation and reactivation of package subscriptions:</p> <ul style="list-style-type: none"> • Suspend – temporarily block data synchronization for a device subscribed to a particular package. • Resume – reactivate a package subscription after it has been suspended. 	Select the box adjacent to the subscription ID of the device, and click either Suspend or Resume .	Messaging DOE-C
Resynchronize	<p>Reactivate subscriptions to a deployed package.</p> <p>If a DOE-C subscription does not respond to the SAP DOE quickly enough, the DOE may mark that subscription's queues as "blocked" and stop sending messages to the DOE-C. Resynchronize to resume communication from the DOE to the DOE-C subscription.</p>	Check the box adjacent to the subscription ID of the device, and click ReSync .	DOE-C

Subscription task	Description	Summary	Package type
Purge	Removes subscriptions that are no longer referenced by any active users.	Select the subscription, click Purge , and then select the criteria.	Messaging UNIFIED (replication subscriptions)

Purging Inactive Package Subscriptions Manually

Purge subscriptions for that have been inactive and remove them from the cache database as well as Sybase Control Center's management view.

Note: Devices for which you have purged subscriptions cannot perform any operations. Only purge those subscriptions that are inactive for a long period of time.

1. In the left navigation pane, expand the **Domains** folder and select a domain.
2. Navigate to the **Packages** folder, and select a subscription.
3. In the right pane, select the **Subscriptions** tab.
4. Select the application payload used: replication or messaging.
5. Search for inactive subscriptions for that payload type:
 - a) From **Search**, select **Number of Inactive Days**.
 - b) Type a positive integer, then click **Go**.
6. Select all inactive subscriptions retrieved, then click **Purge**.
7. Click **OK**. The subscriptions that match the purging criteria are physically deleted from cache database.
8. Repeat the step for the alternate payload protocol in a mixed application environment.

Reviewing MBO History

View or clear the error history of a mobile business object (MBO).

1. In the left navigation pane, expand the **Packages** folder and select the package that contains the MBO you want to view.
2. Select the MBO.
3. In the right administration pane, click the **History** tab.
4. To view MBO data from a specific time period, select a **Start date** and **End date** and click **Go**.
5. Review the following data for the MBO:
 - Data refresh time – the time of this data refresh's failure.

- Failed data refresh counts – the number of failed data refreshes that occurred during the specified time period.
 - Last successful data refresh – the date and time of the most recent successful data refresh of this MBO before this refresh failure.
6. To clean MBO history data, click **Clean**.
Data is removed from the cache database.

Reviewing Operation History

View the error history of a mobile business object (MBO) create, delete, or update operation.

1. In the left navigation pane, expand the **Packages** folder and select the package that contains the MBO operation you want to view.
2. Expand the desired MBO and select the operation for which you want to view the error history: **create**, **delete**, or **update**.
3. In the right administration pane, click the **History** tab.
4. To view operation data from a specific time period, select a **Start date** and **End date** and click **Go**.
5. Review the following data for the operation:
 - Operation replay time – time of this operation replay's failure.
 - Number of failed operation replays– the number of failed operation replays that occurred during the specified time period.
 - Last successful operation replay – the date and time of the most recent successful replay of this operation before this operation replay failure.
6. Click **Clean** to remove operation history data.
The lines are removed from the cache database.

DOE-C Packages

Sybase Mobile Workflow for SAP Business Suite and Sybase Mobile Sales for SAP CRM work with Unwired Platform to make parts of SAP Workflow available on your mobile device using SAP Data Orchestration Engine connector (DOE-C) packages.

DOE-C packages implement the messaging payload protocol, so package management that relate to messaging packages also apply to DOE-C packages. The activities listed here only apply to DOE-C.

Setting the Bulk Load Timeout Property

The Subscribe bulk load timeout property is a package level property targeted to BlackBerry clients for initial server-side subscription operations.

Server-side subscription improves performance and is enabled on the client if the device has a secure digital (SD) memory card enabled. The timeout allows you to set an initial subscription

push timeout period. If the timeout period is reached, Unwired Server sends the database file to the device, whether the initial subscribe is complete or not. The timeout window signals that the device has received sufficient import messages to send the server-built database to the client.

Note: This option is only available for Sybase SAP Data Orchestration Engine Connector (DOE-C) packages.

1. In the left navigation pane of Sybase Control Center, expand the Packages folder and select the package to configure.
2. In the right administration pane, select the **Settings** tab.
3. Set the timeout value. The default value is 3600 seconds.

In addition to the timeout value, you can define the **Subscribe Bulk Load Thread Pool Size** – the maximum number of threads allocated to initial server-side subscription operations. The default value is 5. Setting the thread pool size too high can impact performance. This is a server-side setting that can be set:

- a) In the left navigation pane, expand the **Servers** folder and select a server.
- b) Select **Server Configuration**.
- c) In the right administration pane, select the **General** tab.
- d) From the menu bar, select **Performance Configuration**.
- e) Expand **Show Optional Properties**.
- f) Restart Unwired Server if you change the **Subscribe Bulk Load Thread Pool Size** value for it to take effect.

Checking and Resolving DOE-C User Failures

If the General tab of a DOE-C package displays an invalid user account error for the Error State property, you must resolve the issue by reconfiguring the username and password in the Connection Pool configured for the SAP package connection.

1. In the default domain, expand the **Packages** folder and click the DOE-C package name.
2. Check the Error State property in the **General** tab.
3. Validate the username and password configured, by clicking the **Connection** tab.
4. Correct the user credentials used by editing the corresponding connection pool properties:
 - a) In the navigation pane, click **Connections**.
 - b) In the administration pane, click the **Connections** tab.
 - c) Select the connection for the DOE-C package, then click **Properties**.
 - d) Set the username and password so that it matches the user account credentials.

Note: If you change the username or password property of a DOE-C connection, you must reopen the same dialog and click **Test Connection** after saving. Otherwise the error state of this DOE-C package cannot be cleaned up. If you do not click **Test**

Connection, the username or password is correct, but the error state of the DOE-C package cannot be cleaned up.

Package Subscription Properties

Review information on SAP Data Orchestration Engine connector (DOE-C) package subscriptions in order to manage the synchronization data that device users receive.

Package subscription properties include:

- Application Connection ID – the unique identifier for a user application connection.
- User – the name of the user associated with the application ID.
- Last Server Response Time – the date and time that the last outbound response was sent from Unwired Server to the client.
- Client ID – the device application ID, which identifies the package database for the application.
- Application Name – the name of the device application used by the subscription.
- Status – the current status of a device. The possible values are: Running, Suspended, Pending Activation, Online, Offline, and Expired.
- Packet Dropped – the current packet dropped state of a device. The values are true or false.

Select Advanced to view these properties:

- Subscription ID – the unique identifier of the subscription.
- Logical ID – the unique identifier of a registered device that is generated and maintained by Unwired Server.

Mobile Workflows

Mobile Workflow packages support occasionally connected users and solve the replication and synchronization issues such users present with respect to data concurrency. Workflow packages are similar to other package types in that an administrator must deploy the package to Unwired Server so that it can be configured and made available to client devices.

Enabling and Configuring the Notification Mailbox

Configure the notification mailbox settings that allow Unwired Server to transform e-mail messages into mobile workflows.

The notification mailbox configuration uses a listener to scan all incoming e-mail messages delivered to the particular inbox specified during configuration. When the listener identifies an e-mail message that matches the rules specified by the administrator, it sends the message as a mobile workflow to the device that matches the rule.

Note: Saving changes to the notification mailbox configuration deletes all e-mail messages from the account. Before proceeding with configuration changes, consult your e-mail administrator if you want to back up the existing messages in the configured account.

1. In the left navigation pane, click **Workflows**.
2. In the right administration pane, click **Notification Mailbox**.
3. Select **Enable**.
4. Configure these properties:
 - **Protocol** – choose between POP3 or IMAP, depending on the e-mail server used.
 - **Use SSL** – encrypt the connection between Unwired Server and the e-mail server in your environment.
 - **Server** and **Port** – configure these connection properties so Unwired Server can connect to the e-mail server in your environment. The defaults are localhost and port 110 (unencrypted) or 995 (encrypted).
 - **User name** and **Password** – configure these login properties so Unwired Server can log in with a valid e-mail user identity.
 - **Truncation limit** – specify the maximum number of characters taken from the body text of the original e-mail message, and downloaded to the client during synchronization. If the body exceeds this number of characters, the listener truncates the body text to the number of specified characters before distributing it. The default is 5000 characters.
 - **Poll seconds** – the number of seconds the listener sleeps between polls. During each poll, the listener checks the master inbox for new e-mail messages to process. The default is 60 seconds.
5. If you have added at least one distribution rule, you can click **Test** to test your configuration. If the test is successful, click **Save**.

Configuring a Mobile Workflow Package

Configure mobile workflow properties for you production environment.

Prerequisites

You must deploy a package before you can configure properties for it.

Configuring General Mobile Workflow Properties

Configure general properties for a mobile workflow, including display name and icon. Alter these settings to modify development environment values to production environment equivalents.

1. In the left navigation pane, click **Workflows** and select the mobile workflow for which to configure the properties.
2. In the right administration tab, click **General**.

Only an administrator can change these properties. All others are configured by the mobile workflow developer and cannot be modified.

- **Display name** – sets the name that appears for the mobile workflow package.
 - **Display icon** – scroll and select the icon you want to use for the mobile workflow package.
3. Click Lock/Unlock to lock or unlock a mobile workflow.
A locked mobile workflow cannot be modified or deployed.
 4. Click **Save**.

Configuring Matching Rules

Define the parameters and matching rules that determine whether an e-mail message is a regular e-mail, or a workflow e-mail at runtime.

Prerequisites

The developer must have created an object query and added an E-mail Subscription starting point when the mobile workflow was designed.

Task

When a multiplexer, which a user configures for a Notification mailbox, retrieves e-mails from the e-mail server, the "Matching rules" are used to determine if an e-mail is a regular e-mail or a workflow e-mail. If the "matching rules" match, then the e-mail is processed as a workflow e-mail. The e-mail is processed for "Extraction rules" (this processing is not visible in SCC) to extract values from the e-mail. This determines further processing, such as calling MBO object queries, and so forth. Then, a workflow message is constructed with the necessary data of the object query result, and sent to device(s) according to the "Distribution rules" (which determine the devices to which the message should be sent).

You can configure a matching rule at one of two levels:

- At the inbox level to route e-mails for all mobile workflows
 - At the package level to route e-mails only for a specific mobile workflow
1. In the left navigation pane, click **Workflows > WorkFlowName**.
 2. In the right administration pane, click the **Matching Rules** tab.
 3. Configure matching rules by either:
 - Clicking **Add** to create a new rule, or,
 - Selecting an existing rule name and clicking **Properties**.
 4. In the **Matching Rules** dialog:
 - a) Select the field in the e-mail from which the parameter value is extracted. For example, if you choose From, the parameter value is extracted from the line of the e-mail message that indicates the name of the sender of the message.

You can also select one of the custom parameter values. When registering mobile workflow devices, an administrator can choose one of four device settings. The customer then populates the settings with whatever values they like. Custom parameters can also be set programmatically through Web services.

Note: If you are editing properties of a rule created by a developer, you cannot modify the matching rules.

- b) Choose the type of search expression:
- Equals – the field must exactly match the text in the label.
 - Begins with – the field must begin with the text in the label.
 - Ends with – the field must end with the text in the label.
 - Contains – the text in the label must exist somewhere in the field.
 - Regular expression – search for text that matches the pattern defined by the regular expression. You can create an expression with Boolean operators, groups, or wildcards like "?" or "*". Unwired Platform uses the Boost regular expression engine. See the Boost documentation on regular expression syntax at http://www.boost.org/doc/libs/1_40_0/libs/regex/doc/html/boost_regex/syntax.html.
- c) Configure the text to search against, or define a regular expression in the **Value** field.

Next

Test all new and changed rules to ensure they work as designed.

Testing Configured Matching Rules

Test a new or modified matching rule to ensure it is configured correctly.

1. In the **Matching Rules** tab for a selected mobile workflow package, click **Test**.
2. Populate the fields to create a sample e-mail message against which the rule configuration is tested. Review the results:
 - If a pattern for a corresponding field in the rule matches, the word **Pass** appears adjacent to the field.
 - If a pattern for a corresponding field in the rule does not match, the word **Fail** appears adjacent to the field.
 - Otherwise, **No Rule** appears, indicating that no rule was created for this corresponding e-mail field.

Configuring Context Variables for Mobile Workflow Packages

The administrator can change some of the values of a selected variable, should the design-time value need to change for a production environment.

Which values are configurable depends on whether the developer hard-coded a set of user credentials or used a certificate.

1. In the left navigation pane, click **Workflows** > <**Workflow_Package**>:<**Workflow_Version**>.
2. In the right administration pane, click the **Context Variables** tab.
3. Select the context variable to configure, then click **Modify**.

Context Variable	Description
SupUser	The valid mobile workflow application user name that is used to authenticate the user with the chosen EIS. An administrator must change development/test values to those required for a production environment.
SupUnrecoverableErrorRetryTimeout	After sending a JSON request to Unwired Server, if you receive an EIS code that indicates an unrecoverable error in the response log, the mobile workflow client throws an exception. A retry attempt is made after a retry time interval, which is set to three days by default. Select this property to change the retry time interval.
SupThrowCredentialRequeston401Error	The default is true , which means that an error code 401 throws a <code>CredentialRequestException</code> , which sends a credential request notification to the user's inbox. If this property is set to false , error code 401 is treated as a normal recoverable exception.
SupRecoverableErrorRetryTimeout	After sending a JSON request to Unwired Server, if you receive an EIS code that indicates a recoverable error in the response log, the mobile workflow client throws an exception. A retry attempt is made after a retry time interval, which is set to 15 minutes by default. Select this property to change the retry time interval.
SupPassword	The valid mobile workflow application user password that is used to authenticate the user with the chosen EIS. An administrator must change development/test values to those required for a production environment.
SupPackages	The name and version of the deployed Mobile Workflow package. This cannot be changed.

Context Variable	Description
SupMaximumMessageLength	<p>Use this property to increase the allowed maximum Mobile Workflow message size. Limitations vary depending on device platform:</p> <ul style="list-style-type: none"> • For BlackBerry 5, the limit is 512KB. • For Windows Mobile the limit is 500KB. • For BlackBerry 6 and Android, the limit depends on the memory condition of the device. Large message may result in an out of memory error.

4. In the Context Variable dialog, change the value of the named variable and click **OK**.

Changing Hard Coded User Credentials

The administrator can change hard coded user credentials assigned at design time if the design time value needs to change for a production environment.

1. In the left navigation pane, click **Workflows > <Workflow_Package>:<Workflow_Version>**.
2. In the right administration pane, click the **Context Variables** tab.
3. Select one or both of the variables: SupUser or SupPassword, and click **Modify**.
4. Type the new value and click **OK**.

Adding a Certificate File to the Mobile Workflow Package

The administrator can change the credential certificate assigned at design time.

Note: Sybase recommends that you use Internet Explorer to perform this procedure.

1. In the left navigation pane, click **Workflows > <Workflow_Package>:<Workflow_Version>**.
2. In the right administration pane, click the **Context Variables** tab.
3. Select SupPassword and click **Modify**.
4. To select a new certificate file, click **Browse** to find and upload a production version of the certificate.
The read-only values of SupUser, SupCertificateSubject, SupCertificateIssuer, SupCertificateNotAfter, SupCertificateNotBefore change to reflect values of the new certificate and password you set.
5. Click **OK**.

Assigning and Unassigning Mobile Workflows

Assign mobile workflow packages to make them available to a device user. Unassign them when a package is no longer required.

1. In the left navigation pane of Sybase Control Center, click **Workflows > <Mobile_WorkFlow_Package>**.
2. In the right administration pane, click the **Application Connections** tab.
3. Locate the device to assign a mobile workflow package to, then:
 - a) Click **Assign Workflow**.
 - b) List the activation users to assign the mobile workflow package to.

By default, no users are listed in this window. Search for users by selecting the user property you want to search on, then selecting the string to match against. Click **Go** to display the users.
 - c) Select the user or users from the list to which to assign the mobile workflow package.
 - d) Click **OK**.
4. To unassign a mobile workflow package, select the User and click **Unassign Workflow**.

Checking Mobile Workflow Users and Queues

Check mobile workflow application users and review pending activities for a mobile workflow application.

1. In the left navigation pane, expand the **Workflows** folder and select the mobile workflow you want to administer.
2. To check mobile workflow users:
 - a) In the right administration pane, select the **Application Connections** tab.
 - b) Review data about mobile workflow device users:
 - User – the name of the user that activates the device.
 - Application Connection ID – the unique identifier for a user application connection.
 - Errors – the total number of errors on the device.
 - Transform Items – the total number of items in the transform queue. The transform queue contains items that Unwired Server has transformed from e-mail messages into mobile workflow messages to be sent to clients.
 - Response Items – the total number of items in the response queue. The response queue contains mobile workflow messages that are sent from the device to Unwired Server.
 - Device Number – the unique identifier for a registered mobile device that is generated and maintained by Unwired Server.
 - Transform Queue Status – the current status of the transform queue: active, awaiting credentials, or awaiting retry.
 - Response Queue Status – the current status of the response queue: active, awaiting credentials, or awaiting retry.
3. To view pending activities for a mobile workflow:
 - a) Select the **Queue Items** tab.

Deploy

- b) Review data about pending mobile workflow activities:
 - User – the name of the user that activates the device.
 - Queue Type – the type of mobile workflow queue: response or transform.
 - Application Connection ID – the unique identifier for a user application connection.
 - Device Number – the unique identifier for a registered mobile device that is generated and maintained by Unwired Server.
 - Queue ID – the unique identifier of the queued item.
 - State – the status of the mobile workflow queue: active, awaiting credentials, or awaiting retry.
 - Creation Date – the date the queue item was created.
 - Retry Date – the date that the processing of the queue item is scheduled to be retried (if applicable).
4. To manage the mobile workflow queue in the event of non-recoverable errors:
 - a) Select the **Queue Items** tab.
 - b) Identify a workflow queue item that requires you to unblock it or delete it.

Errors affecting workflow queue items are either recoverable (where a retry is applicable) or unrecoverable/unknown (where no automatic retry occurs, or there is very long retry interval).

To recover from a long retry interval, an administrator can unblock a queue currently in retry state, so the next work schedule can pick up the blocked item immediately, instead of waiting for the retry timeout.
 - c) Select one or more of the queue items from the same queue type (the queue type for all the selected items must be either Transform or Retry).
 - d) Select one of the following actions:
 - **Delete** – deletes the selected workflow queue item(s).
 - **Unblock** – unblocks the selected workflow queue item(s) that are currently in a retry state.
 - e) Click **OK** to confirm the action.

Deploying a Mobile Workflow Package

Use the Deploy wizard to make mobile workflow packages available on the Unwired Server.

1. In the left navigation pane, click **Workflows**.
2. From the **General** tab, click **Deploy**.
3. Click **Browse** to locate the Mobile Workflow package.
4. Select the file to upload and click **Open**.
5. Select the deployment mode:
 - New – generates and deploys the mobile workflow package and its files for the first time.

If the uploaded file does not contain a Mobile Workflow, or a Mobile Workflow with the same name and version is already deployed to Unwired Server, you see an error message.

- Update – updates the existing mobile workflow package with the newly generated mobile workflow package and its files before deploying. The previous version of the package remains on the server
- Replace – replaces any pre-existing mobile workflow packages while preserving any user assignments.

The package is added to the list of deployed packages, which are sorted by Display Name.

Next

Configure the package if you want the deployed package to have a different set of properties in a production environment.

Deploy

Monitor

Monitor availability status, view system and performance statistics, and review system data that allow administrators to diagnose the health and security of the runtime.

Monitored operations include security, replication-based synchronization, messaging-based synchronization, messaging queue, data change notification, device notification, package, user, and cache activity. These aspects of monitoring are important to ensuring that the required data is collected.

The critical aspects of monitoring include:

1. Setting up a monitoring configuration. A monitoring configuration sets the server behavior for writing data to database, automatic purge, and data source where the monitoring data is stored.

A default configuration is created for you, however you will likely want to customize this configuration for your environment. By default, monitoring data is flushed every 5 minutes. In development and debugging scenarios, you may need to set the flush behavior to be immediate. Set the **Number of rows** and **Batch size** properties to a low number. You can also disable flush, which results in immediately persisting changes to monitoring database. If you are setting up immediate persistence in a production environment, you may experience degraded performance. Use persistence with caution.

2. Creating a monitoring profile. A monitoring profile defines one or more domains and packages that need to be monitored.

You can either use the **default** profile to capture monitoring data for all packages in all domains or create specific profiles as required. Otherwise, disable the **default** profile or modify it as needed.

3. Reviewing the captured data. An administrator can review monitoring data (current, historical, and performance statistics) from Sybase Control Center.

Use the monitoring tabs to filter the data by domain, package, and time range. You can also export the data into a CSV or XML file and then use any available reporting or spreadsheet tool to analyze the data.

Monitoring Usage

Monitoring information reflects current and historical activity, and general performance during a specified time period.

Monitoring allows administrators to identify key areas of weakness or periods of high activity in the particular area they are monitoring. Access to this data helps administrators make decisions about how to better configure the application environment to achieve a higher level of performance.

Monitor

The historical data is preserved in the monitor database. Performance data (KPIs for Replication, Messaging, Package Statistics, User Statistics, and Cache Statistics) for the specified time period is calculated upon request using the historical data available for that period. If monitoring data is purged for that time period, the performance data calculations will not factor in that data. It is recommended to purge monitoring data after putting in place mechanisms to export the required historical and/or performance data as needed. By default, monitoring data is automatically purged after seven days.

Also note that the processing times are calculated based on the time the request (or message) arrives on the server, and the time it took to process the request (or message) on the server. The client-side time (request origin time, and time taken to deliver to the server) are not factored into that data.

System Monitoring Overview

(Not applicable to Online Data Proxy) The goal of monitoring is to provide a record of activities and performance statistics for various elements of the application. Monitoring is an ongoing administration task.

Use monitoring information to identify errors in the system and resolve them appropriately. This data can also be shared by platform and domain administrators by exporting and saving the data to a .CSV or .XML file.

The platform administrator uses Sybase Control Center to monitor various aspects of Unwired Platform. Monitoring information includes current activity, historical activity, and general performance during a specified time period. You can monitor these components:

- Security log
- Replication synchronization
- Messaging synchronization
- System messaging queue status
- Data change notifications
- Device notifications (replication)
- Package statistics (replication and messaging)
- User-related activity
- Cache activity

To enable monitoring, platform administrators must set up a monitoring database, configure a monitoring data source or create a new one, and set up monitoring database flush and purge options. By default the installer created a monitoring database, however you can use another one if you choose.

To control monitoring, platform administrators create monitoring profiles and configurations, which define the targets (domains and packages) to monitor for a configured length of time. A default monitoring profile is created for you by the installer. Monitoring data can be deleted by the platform administrator as needed.

Table 17. System monitoring tasks

Task	Frequency	Accomplished by
Create and enable monitoring profiles	One-time initial configuration with infrequent tuning as required	Sybase Control Center for Unwired Platform with the Monitoring node
Enable domain logging	One-time setup with infrequent configuration changes, usually as issues arise	Sybase Control Center for Unwired Platform with the Domains > <DomainName> > Log node.
Review current/historical/performance metrics	Routine	Sybase Control Center for Unwired Platform with the Monitoring node
Identify performance issues	Active	Sybase Control Center for Unwired Platform with the Monitoring node
Monitor application and user activity to check for irregularities	Active	Sybase Control Center for Unwired Platform with the Monitoring node
Troubleshoot irregularities	Infrequent	Reviewing various platform logs
Purge or export data	On demand	Sybase Control Center for Unwired Platform with the Monitoring node

Monitoring Configuration

The monitoring configuration identifies the monitoring database endpoint and determines how long data is stored in the database.

The configurable monitoring properties are:

- Auto-purge – configures an automatic purge of the monitoring database to occur on a regular basis.
- Flush threshold – determines how often monitoring data is flushed from the server memory for storage in the monitoring database.
- Flush batch size – divides flushed data into smaller segments, rather than saving all data together according to the flush threshold parameters.
- Monitor database endpoint – sets the database to be used for storage of monitoring data.

Configuring Monitoring Performance Properties

Configure auto-purge, flush threshold, and flush batch size settings to determine how long monitoring data is retained, and set a monitoring database to configure where data is stored.

Prerequisites

Depending on the expected level of monitoring activity, ensure that the monitoring database is adequately prepared to store monitoring data.

Task

1. In the left navigation pane of Sybase Control Center, select **Monitoring**.
2. In the right administration pane, select the **General** tab.
3. Click **Configuration**.
4. Configure auto purge settings.

Auto purge clears obsolete data from the monitoring database once it reaches the specified threshold.

- a) Select **Enable auto purge configuration** to activate auto purge functionality.
- b) Enter the length of time (in days) to retain monitoring data before it is purged.

5. Configure flush threshold settings.

The flush threshold indicates how often data is flushed from memory to the database. This allows you to specify the size of the data saved in memory before it is cleared. Alternately, if you do not enable a flush threshold, data is automatically written to the monitoring database as it is captured.

- a) Select **Enable flush threshold configuration** to activate flush threshold functionality.
- b) Select one of:

- **Number of rows** – monitoring data that surpasses the specified number of rows is flushed from memory. Enter the desired number of rows adjacent to **Rows**. The default is 100.
- **Time interval** – monitoring data older than the specified time interval is flushed from memory. Enter the desired duration adjacent to **Minutes**. The default is 5.
- **Either rows or time interval** – monitoring data is flushed from memory according to whichever value is reached first: either the specified number of rows or the specified time interval. Enter the desired rows and duration adjacent to **Rows** and **Minutes**, respectively.

6. If you enabled a flush threshold, enter a **Flush batch row size** by specifying the size of each batch of data sent to the monitoring database. The row size must be a positive integer. The batch size divides flushed data into smaller segments, rather than saving all data together according to the flush threshold parameters. For example, if you set the flush threshold to 100 rows and the flush batch row size to 50, once 100 rows are collected in the

monitoring console, the save process executes twice; data is flushed into the database in two batches of 50 rows. If the flush threshold is not enabled, the flush batch row size is implicitly 1.

Note: By default, the monitoring database flushes data every 5 minutes. Alternatively, you can flush data immediately by removing or decreasing the default values, but doing so impacts performance and prevents you from using captured data.

7. Optional. To change the data source, select an available database from the **Monitor database endpoint** drop down list.

Available databases are those with a JDBC server connection type created in the "default" domain. To create a new monitor database, a platform administrator must set up a database by running the appropriate configuration scripts and creating a server connection for the database in the default domain. The database server connection then appears as an option in the Monitor Database Endpoint drop down list.

8. Click **OK**.

Monitoring Profiles

Monitoring profiles specify a monitoring schedule for a particular group of packages. These profiles let administrators collect granular data on which to base domain maintenance and configuration decisions.

A default monitoring profile is automatically created in disabled state on Unwired Server. Administrators can enable or remove the default profile, and enable one or more new monitoring profiles as required.

The same monitoring schedule can be applied to packages across different domains; similarly, you can select individual packages for a monitoring profile.

Note: Properties you configure for an Unwired Server are cluster-affecting. Therefore, to make sure they are propagated correctly, Sybase recommends that you set them only on a primary cluster server.

Creating and Enabling a Monitoring Profile

Specify a monitoring schedule for a group of packages.

Prerequisites

Depending on the expected level of monitoring activity, ensure that the monitoring database is adequately prepared to store monitoring data.

Task

1. In the left navigation pane, select **Monitoring**.

Monitor

2. In the right administration pane, select the **General** tab.
3. Click **New** to create a monitoring profile.
4. Enter a name for the new profile.
5. Select the **Domains and Packages** tab and choose packages to be monitored according to these options:
 - Monitor all domains and packages – select **All Domains and Packages**.
 - Monitor all packages from one or more domains – select a domain, then click **Select All Packages**. Perform this step for each domain you want to monitor.
 - Monitor specific packages from one or more domains – select a domain, then select the particular packages you want to monitor from that domain. Perform this step for each domain you want to monitor.
6. Select **View my selections** to view the packages you selected for the monitoring profile. Unselect this option to return to the package selection table.
7. Select **Enable after creation** to enable monitoring for the selected packages immediately after you create the profile. By default, this option is selected. Unselect this option to enable the monitoring profile later.
8. On the **Schedule** tab, select a schedule to specify when monitoring takes place:
 - **Always On** – this schedule requires no settings. Package activity is continually monitored.
 - **Run Once** – specify a length of time during which monitoring occurs, in either minutes or hours. Package activity is monitored for the duration specified for one time only.
 - **Custom** – specify start and end dates, start and end times, and days of the week. Package activity is monitored according to the time frame specified. See *Setting a Custom Monitoring Schedule*.
9. Click **OK**.

A status message appears in the administration pane indicating the success or failure of profile creation. If successful, the profile appears in the monitoring profiles table.
10. To enable a profile that you did not enable during creation, select the monitoring profile and click **Enable**.

Setting a Custom Monitoring Schedule

Customize the monitoring schedule for packages within a monitoring profile. Setting a custom schedule is the most flexible option; monitoring information is provided according to the time frame you specify.

Prerequisites

Begin creating a monitoring profile in the New Monitor Profile dialog.

Task

1. In the New Monitor Profile dialog, select the **Schedule** tab.
 2. Select **Custom** as the monitoring schedule criteria.
 3. To set a range to control which days the custom schedule runs, configure a start date and time, end date and time, or day of week (if applicable).
 - Select **Start Date** to set a date for when monitoring of package activity begins. To be more specific, you can also enter a **Start Time**. In this case, monitoring cannot begin until a given time on a given day has been reached.
 - Select **End Date** to set a date that ends the monitoring of package activity. To be more specific, you can also enter an **End Time**.
 - Select the days of the week that package monitoring runs. This means that for the days you select, the schedule runs every week on the day or days you specify.
- If you do not indicate a time frame, Unwired Server uses the default custom schedule, which is equivalent to Always On monitoring.
4. Click **OK**.

Monitoring Data

Monitoring data is aggregated in the Monitoring node of Unwired Server and organized by activity, including security, replication-based synchronization, messaging-based synchronization, messaging queue, data change notifications, device notifications, packages, users, and cache. The data for each activity is further broken down into current, historical, and performance-related information. View data for each monitored activity to track the performance and health of the system.

You can selectively view data accrued during a specific time period to see a snapshot of system performance during specific periods. The export function allows you to save data to a file outside of Sybase Control Center for reference or logging purposes.

Reviewing System Monitoring Data

Review data for monitored activities. The monitoring data is retrieved according to the specified time range. Key Performance Indicators (KPIs) are also calculated for the specified time range.

1. In the left navigation pane, select **Monitoring**.
2. In the right administration pane, select one of the following tabs according to the type of monitoring data you want to view:
 - **Security Log**
 - **Replication**

Monitor

- **Messaging**
- **Queue**
- **Data Change Notifications**
- **Device Notifications**
- **Package Statistics**
- **User Statistics**
- **Cache Statistics**

Purging Monitoring Data

Clear old data from the monitoring database.

Using the Purge function in Sybase Control Center allows you to perform an ad hoc purge of monitoring data.

1. In the left navigation pane, select **Monitoring**.
2. In the right administration pane, select the **General** tab.
3. Click **Purge**.
4. Indicate the time period for which you want to delete data by specifying a **Start Date**, **Start Time**, **End Date**, and **End Time**.

All monitoring data collected from the start time and date to the end time and date is deleted from the database. If you do not specify a start date, all data acquired prior to the end date and time is purged. Similarly, if you do not specify an end date, all data collected from the start date until the present time is purged. To save the data to file before purging it, see *Exporting Monitoring Data*.

5. Click **OK**.

A status message appears in the right administration pane indicating that the data purge was successfully completed.

Exporting Monitoring Data

Save a segment of monitoring data to a location outside of the monitoring database. Export data to back up information, particularly before purging it from the database, or to perform closer analysis of the data in a spreadsheet application.

This option is especially useful when you need to share monitoring data with other administrators and tenants. Since this task can be time-consuming, depending upon the size of the data being exported, Sybase recommends that you export the data in segments or perform the export at a time when Sybase Control Center is not in use.

Note: The time taken to export the requested data is dependent on the time range specified, and the amount of data in the monitoring database. If the export is taking too long, or the user interface is blocked for too long, another option is to export the monitoring data using the management APIs provided for exporting monitoring data. Please refer the *Developer Guide: Unwired Server Management API* for further details.

1. In the left navigation pane, select **Monitoring**.
 2. In the right administration pane, select the tab corresponding to the monitoring data you want to view.
 3. Perform a search using the appropriate criteria to obtain the desired monitoring data.
 4. Click **Export**.
 5. Select a file type for the exported data (CSV or XML), and click **Next**.
 6. Click **Finish**.
 7. In the file browser dialog, select a save location and enter a unique file name.
 8. Click **OK**.
- All monitoring data retrieved by the search is saved to the file you specify in step 7.

Searching Monitoring Data

Filter monitoring data according to a specified date and time range.

Filter options vary depending upon the type of monitoring data you search.

1. In the left navigation pane, select **Monitoring**.
2. In the right administration pane, select the tab corresponding to the monitoring data you want to view.
3. In the Search pane, indicate the time period for which you want to view data by specifying a date range to search within (that is, Start Date, Start Time, End Date, and End Time) .

Note: You do not need to specify a time period if you are performing a search on Current date.

4. Filter the system components to include in the search.
 - a) Select **Show current filter**.
 - b) Specify the components to include in your search.If a domain or package does not appear in the search list (for example, if it has been deleted), enter the name and click **Add**.
5. From the **Sort By** drop-down list, select a category by which to sort search results.

Note: This field is disabled for some categories.

6. To view the domains and packages you included in the search, select **View My Selections**.
7. Click **Retrieve**.

Monitoring data for the time period specified appears in the administration console.

Monitoring Data Categories

Monitoring data is organized according to object type, allowing administrators to perform focused data analysis on specific activities and Unwired Platform components. Current,

Monitor

historical, and performance-based statistics facilitate user support, troubleshooting, and performance tracking for individual application environments.

The replication and messaging categories are the primary sources of data relating to application environment performance. The remaining tabs present detailed monitoring data that focuses on various aspects of replication-based applications, messaging-based applications, or both.

Security Log Statistics

The security log reflects the authentication history of users either across the cluster, or filtered according to domain, during a specified time period. These statistics allow you to diagnose and troubleshoot connection or authentication problems on a per-user basis. Security log monitoring is always enabled.

User security data falls into these categories:

Category	Description
User	The user name
Security Configuration	The security configuration to which the device user belongs
Time	The time at which the authentication request took place
Result	The outcome of the authentication request: success or failure
Application Connection ID	The application connection ID associated with the user
Package	The package the user was attempting to access
Domain	The domain the user was attempting to access

Replication Statistics

Replication statistics reflect replication synchronization activity for monitored packages. Current statistics monitor the progress of real-time synchronizations, while historical statistics present data from completed synchronizations on a per-package basis. Performance monitoring uses key performance indicators to produce data about synchronization efficiency.

Through statistics that report on the duration and scope of synchronizations, as well as any errors experienced during synchronization, replication monitoring allows you to identify the rate at which synchronizations happen during specified time periods, which users synchronize data, and which mobile business objects are affected.

Current Replication Statistics

Current statistics for replication synchronization provide real-time information about in-progress synchronizations.

Unwired Server monitors replication requests using these statistical categories:

Category	Description
Application ID	The ID associated with the application.
Package	The package name.
Phase	The current synchronization activity: upload or download. During the upload phase, a client initiates operation replays to execute mobile business object (MBO) operations on the back-end system. During the download phase, a client synchronizes with Unwired Server to receive the latest changes to an MBO from the back-end system.
Entity	During the download phase, the name of the MBO with which the client is synchronizing. During the upload phase, the name of the operation that the client is performing.
Synchronization Start Time	The date and time that the synchronization request was initiated.
Domain	The domain to which the package involved in synchronization belongs.
Application Connection ID	The ID number of the connection participating in the synchronization.
User	The name of the user associated with the device ID.

Replication History Statistics

Historical data for replication-based synchronization consists of past synchronization details for monitored packages.

The summary view provides general information, whereas the detail view presents a more specific view of all request events during each synchronization; each row of data corresponds to a synchronization request from the client in the time frame you define:

- Click either **Details** to see more granular information on each synchronization request, or select the **Detail** option to see all synchronization request details. Detail view allows you to look at the individual messages that make up the summary view.
- Select **Summary** to see aggregated details by domain, package, and user about past synchronization events for the defined time frame.

Table 18. Detail view information

Synchronization element	Description
Application ID	The ID number associated with an application.
Package	The package name.

Synchronization element	Description
Application Connection ID	The ID number of the connection used in a synchronization request.
User	The user associated with the device ID.
Phase	The sync activity that occurred during this part of synchronization: upload or download. During the upload phase, a client initiates operation replays to change an MBO. During the download phase, a client synchronizes with Unwired Server to receive the latest changes to an MBO.
Entity	During download, the name of the MBO that the client is synchronizing with. During upload, the operation that the client is performing: create, update, or delete.
Total Rows Sent	The total number of rows sent during package synchronization. This data type is not supported at the MBO level.
Bytes Transferred	The amount of data transferred during the synchronization request.
Start Time	The date and time that the synchronization request was initiated.
Finish Time	The date and time that this part of synchronization completed.
Error	The incidence of errors during this request: true or false.
Domain	The domain to which the package involved in synchronization belongs.

Table 19. Summary view information

Category	Description
Application ID	The ID number associated with an application.
User	The name of the user associated with the device ID.
Package	The package name.
Total Rows Sent	The total number of rows sent during package synchronization.
Total Operation Replays	The total number of operation replays performed by clients during synchronization.
Total Bytes Sent	The total amount of data (in bytes) downloaded by clients from Unwired Server during synchronization.

Category	Description
Total Bytes Received	The total amount of data (in bytes) uploaded to Unwired Server by clients during synchronization.
Start Time	The date and time that the synchronization request was initiated.
Total Synchronization Time	The amount of time taken to complete the synchronization.
Total Errors	The total number of errors that occurred for the package during synchronization.
Domain	The domain to which the package involved in synchronization belongs.

Replication Performance Statistics

Replication performance statistics consist of key performance indicators (KPIs) that reflect the overall functioning of the application environment.

Performance monitoring highlights key totals and identifies average, minimum, and maximum values for primary activities. These calculations are dynamic, and are based on the data currently available in monitoring database for the specified time period.

All values in this table (totals, averages, maximums, minimums) apply to the specific time period you indicate:

KPI	Description
Total Distinct Package Synchronization	The total number of packages subject to synchronization.
Total Distinct Users	The total number of users who initiated synchronization requests. This value comprises only individual users, and does not count multiple synchronizations requested by the same user.
Average/Minimum/Maximum Sync Time	The average, minimum, or maximum amount of time Unwired Server took to finish a complete synchronization.
Time at Minimum/Maximum Sync Time	The time of day at which the shortest or longest synchronization completed.
Package with Minimum/Maximum Synchronization Time	The name of the package and associated MBO with the shortest or longest synchronization time.
Average/Minimum/Maximum MBO Rows Per Synchronization	The average, minimum, or maximum number of MBO rows of data that are downloaded when synchronization completes.

KPI	Description
Average/Minimum/Maximum Operation Replays per Sync (records received)	The average, least, or greatest number of operation replays per synchronization received by Unwired Server from a client.
Total Bytes Sent	The total number of bytes downloaded by clients from Unwired Server.
Total Bytes Received	The total number of bytes uploaded from clients to Unwired Server.
Total Operation Replays	The total number of operation replays performed on the EIS.
Total Errors	The total number of errors that took place across all synchronizations.
Average/Minimum/Maximum Concurrent Users	The average, least, or greatest number of users involved in concurrent synchronizations.
Time at Minimum/Maximum Concurrent Users	The time at which the least or greatest number of users were involved in concurrent synchronizations.

Messaging Statistics

Messaging statistics report on messaging synchronization activity for monitored packages.

- Current monitoring data tracks the progress of messages from device users presently performing operation replays or synchronizing MBOs.
- Historical data reveals statistics indicating the efficiency of completed transactions.
- Performance monitoring provides an overall view of messaging payload activity intended to highlight areas of strength and weakness in the application environment.

Messaging historical data captures messages such as login, subscribe, import, suspend, resume and so on. The Import type message is a data payload message from server to client (outbound messages), while rest of the messages (login, subscribe, replay, suspend, resume) are sent from the client to server (inbound messages).

Current Messaging Statistics

Current statistics for messaging synchronization provide real-time information about in-progress synchronizations. Because messaging synchronizations progress rapidly, there is typically little pending messaging data available at any given time.

Unwired Server monitors messaging requests using these categories:

Category	Description
Application ID	The ID associated with the application.
Package	The package name.

Category	Description
Message Type	The type of message sent by the client to Unwired Server, indicating the current sync activity; for example, import, replay, subscribe, suspend, resume, and so on.
Entity	During the import process, the name of the mobile business object (MBO) with which the client is synchronizing. During replay, the operation that the client is performing. For all other message types, the cell is blank.
Start Time	The date and time that the initial message requesting synchronization was sent by the client.
Domain	The domain to which the package involved in synchronization belongs.
Application Connection ID	The ID number of the application participating in the synchronization.
User	The name of the user associated with the device ID.

Messaging History Statistics

Historical data for messaging synchronization consists of past synchronization details for monitored packages.

The summary view provides general information, whereas the detail view presents a more specific view of all request events during each synchronization; each row of data corresponds to a synchronization request from the client in the time frame you define:

- Click either **Details** to see more granular information on each synchronization request, or select the **Detail** option to see all synchronization request details. Detail view allows you to look at the individual messages that make up the summary view.
- Select **Summary** to see aggregated details by domain, package, and user about past synchronization events for the defined time frame.

Table 20. Detail view information

Data type	Description
Application ID	The ID number associated with an application.
Package	The package name.
Application Connection ID	The ID number of the connection that participated in the synchronization request.
User	The name of the user associated with the device ID.

Data type	Description
Message Type	The type of message sent by the client to Unwired Server, indicating the sync activity; for example, import, replay, subscribe, suspend, resume, and so on.
Entity	During the import process, the name of the mobile business object (MBO) that the client is synchronizing with. During replay, the operation that the client is performing. For all other message types, the cell is blank.
Payload Size	The size of the message (in bytes).
Start Time	The date and time that the message for this sync request is received.
Finish Time	The date and time that the message for this sync request is processed.
Processing Time	The total amount of time between the start time and the finish time.
Error	The incidence of errors during this request; either true or false.
Domain	The domain to which the package involved in synchronization belongs.

Table 21. Summary view information

Category	Description
Application ID	The ID number associated with an application.
User	The name of the user associated with the device ID
Package	The package name
Total Messages Sent	The total number of messages sent by Unwired Server to clients during synchronization
Total Messages Received	The total number of messages received by Unwired Server from clients during synchronization
Total Payload Size Sent	The total amount of data (in bytes) downloaded by clients from Unwired Server during synchronization
Total Payload Size Received	The total amount of data (in bytes) uploaded to Unwired Server by clients during synchronization
Total Operation Replays	The total number of operation replays performed by clients during synchronization
Last Time In	The date and time that the last inbound request was received

Category	Description
Last Time Out	The date and time that the last outbound response was sent
Subscription Commands Count	The total number of subscription commands sent during synchronization; for example, subscribe, recover, suspend, and so on
Total Errors	The number of errors that occurred for the package during synchronization
Domain	The domain to which the package involved in synchronization belongs

Messaging Performance Statistics

Messaging performance statistics consist of key performance indicators (KPIs) that reflect the overall functioning of the application environment.

Performance monitoring highlights key totals and identifies average, minimum, and maximum values for primary activities. These calculations are dynamic, and are based on the data currently available in monitoring database for the specified time period.

All values in this table (totals, averages, maximums, minimums) apply to the specific time period you indicate:

KPI	Description
Total Messages	The total number of messages sent between the server and clients during synchronization.
Total Distinct Devices	The total number of devices involved in synchronization. This total includes the same user multiple times if he or she has multiple devices. The value comprises individual devices, and does not count multiple synchronizations requested by the same device.
Total Distinct Users	The total number of users who initiated synchronization requests. This value comprises individual users, and does not count multiple synchronizations requested by the same user if he or she uses multiple devices.
Average/Minimum/Maximum Concurrent Users	The average, minimum, or maximum number of users involved in simultaneous synchronizations.
Time at Minimum/Maximum Concurrent Users	The time at which the greatest or least number of users were involved in concurrent synchronizations.
Average/Minimum/Maximum Processing Time	The average, minimum, or maximum amount of time Unwired Server took to respond to a sync request message.

KPI	Description
Time at Minimum/Maximum Message Processing Time	The time of day at which the shortest or longest message processing event completed.
MBO for Maximum/Minimum Message Processing Time	The name of the package and associated mobile business object (MBO) with the shortest or longest message processing time.
Average/Minimum/Maximum Message Size	The average, smallest, or largest message sent during synchronization.
Total Inbound Messages	The total number of messages sent from clients to Unwired Server.
Total Outbound Messages	The total number of messages sent from Unwired Server to clients.
Total Operation Replays	The total number of operation replays performed by clients on MBOs.
Total Errors	The total number of errors that took place across all synchronizations.
Average/Minimum/Maximum Concurrent Users	The average, least, or greatest number of users involved in concurrent synchronizations.

Note: Reporting of KPIs related to concurrent users is based on a background task that takes a periodic snapshot of the messaging activities. Depending on the nature and length of the processing of a request, the background snapshot may not always see all the requests.

Messaging Queue Statistics

Messaging queue statistics reflect the status of various messaging queues. The data does not reveal any application-specific information, but provides a historical view of messaging activities that communicates the efficiency of messaging-based synchronization, as well as the demands of device client users on the system.

Based on this data, administrators can calculate the appropriate inbound and outbound message queue counts for the system (configurable in the Server Configuration node of Sybase Control Center).

Messaging Queue Status

Messaging queue status data provides historical information about the processing of messaging-based synchronization requests by Unwired Server. The data indicates areas of high load and times of greatest activity. This data can help administrators decide how to handle queue congestion and other performance issues.

These key indicators monitor messaging queue status:

Statistic	Description
Name	The name of the messaging queue.
Current Queued Items	The total number of pending messages waiting to be processed by Unwired Server.
Average/Minimum/Maximum Queue Depth	The average, minimum, or maximum number of queued messages. For minimum and maximum queue depth, this value is calculated from the last server restart.
Time at Minimum/Maximum Queue Depth	The time and date at which the queue reached its minimum or maximum depth.
Type	The direction of message flow: inbound or outbound.
Total Messages	The total number of messages in the queue at one point since the last server reboot.
Bytes Received	The total number of bytes processed by the queue since the last server reboot.
Last Activity Time	The time at which the most recent message was added to the queue since the last server reboot.

Data Change Notification Statistics

Data change notification (DCN) statistics monitor notifications that are received by Unwired Server from the enterprise information server. Specifically, DCN monitoring reports which packages and sync groups are affected by notifications, and how quickly these are processed by the server.

Monitoring DCN statistics allows you to troubleshoot and diagnose performance issues if, for example, the cache is not being updated quickly enough. These statistics help to identify which packages took longest to process data changes, as well as times of peak performance or strain on the system.

Data Change Notification History Statistics

Historical information for data change notifications (DCNs) consists of past notification details for monitored packages. Detailed data provides specific information on past notification activity for packages, and identifies which server data was affected.

Details about past notification events are organized into these categories:

Category	Description
Domain	The domain to which the package affected by the DCN belongs.
Package	The name of the package containing data changes.

Category	Description
MBO	The name of the MBO to which the notification applied.
Notification Time	The date and time that Unwired Server received the DCN.
Processing Time	The time that Unwired Server used to process the DCN.

Data Change Notification Performance Statistics

Data change notification (DCN) performance statistics consist of key performance indicators that reflect the efficiency of notification processing by Unwired Server.

Performance monitoring highlights key totals and identifies average, minimum, and maximum values for primary activities. These calculations are dynamic, and are based on the data currently available in monitoring database for the specified time period.

All values in this table (totals, averages, maximums, minimums) apply to the specific time period you indicate:

Key performance indicator	Description
Total Notifications	The total number of notifications sent by the enterprise information system to Unwired Server.
Average/Minimum/Maximum Processing Time	The average, minimum, or maximum amount of time Unwired Server took to process a DCN.
Time at Minimum/Maximum Message Processing Time	The time of day at which the shortest or longest DCN processing event completed.
Time of Last Notification Received	The time at which the most recent DCN was received by Unwired Server.
MBO with Minimum/Maximum Notification Processing Time	The name of the package and associated mobile business object (MBO) with the shortest or longest notification processing time.

Device Notification Statistics

Device notification statistics provide data about the occurrence and frequency of notifications sent from Unwired Server to replication synchronization devices.

Historical device notification monitoring reports on the packages, synchronization groups, and devices affected by replication payload synchronization requests in a given time frame. Performance-related device notification data provides a general indication of the efficiency of notification processing and the total demand of synchronization requests on the system.

Device Notification History Statistics

Historical information for device notifications provides specific information on past device notifications, indicating which packages, synchronization groups, and devices were involved in synchronization requests.

Details about past device notification events fall into these categories:

Category	Description
Application ID	The ID associated with the application.
Domain	The domain to which the package affected by the device notification belongs.
Package	The name of the package containing data changes.
Synchronization group	The synchronization group that the package belongs to.
Application Connection ID	The ID number of the connection participating in the synchronization request.
Generation time	The date and time that Unwired Server generated the device notification.
User	The name of the user associated with the device ID.

Device Notification Performance Statistics

Device notification performance statistics provide a general indication of the efficiency of notification processing and the total demand of synchronization requests on the system.

Performance monitoring highlights key totals and identifies average, minimum, and maximum values for primary activities. These calculations are dynamic, and are based on the data currently available in monitoring database for the specified time period.

All values in this table (totals, averages, maximums, minimums) apply to the specific time period you indicate:

KPI	Description
Synchronization Group for Maximum Notifications	The synchronization group for which the maximum number of notifications were sent.
Package for Maximum Notifications	The package for which the greatest number of device notifications were sent.
Total Notifications	The total number of device notifications sent from Unwired Server to devices.

KPI	Description
Total Distinct Users	The total number of users that received device notifications. This value comprises only individual users, and does not count multiple synchronizations requested by the same user.
Total Distinct Devices	The total number of devices that received device notifications. This is distinct from Total Distinct Users, because a single user name can be associated with multiple devices.
Enabled Subscriptions	The total number of replication subscriptions for which notifications are generated.
Time at Last Notification	The time at which the last device notification was sent by Unwired Server.
Outstanding Subscriptions	The total number of replication subscriptions, both enabled and disabled.

Package Statistics

Package statistics reflect response times for replication-based and messaging-based synchronization packages.

This type of monitoring uses key performance indicators to provide data on the efficiency of response by Unwired Server to synchronization requests. These calculations are dynamic, and are based on the data currently available in monitoring database for the specified time period.

Replication Package Statistics

Replication package statistics consist of key performance indicators (KPIs) that reflect the overall function of the application environment at the cluster or domain level. The statistics highlight key totals and identify average, minimum, and maximum values for primary activities.

These key indicators monitor replication packages:

Note: These KPIs are not applicable at the MBO level.

- Total Bytes Received
 - Total Bytes Sent
 - Total Operation Replays
-

KPI	Description
Total Devices	The total number of devices involved in synchronization. This total includes the same user multiple times if he or she has multiple devices. The value comprises individual devices, and does not count multiple synchronizations requested by the same device.
Total Rows Sent	The total number of rows sent during package synchronization.
Total Rows Received	The total number of rows received during package synchronization.
Total Errors	The total number of errors that took place across all synchronizations.
Total Bytes Received	The total number of bytes uploaded from clients to Unwired Server.
Total Bytes Sent	The total number of bytes downloaded by clients from Unwired Server.
Average/Minimum/Maximum Synchronization Time	The average, minimum, or maximum amount of time Unwired Server took to finish a complete synchronization.
Time at Minimum/Maximum Synchronization Time	The time at which the shortest or longest synchronization completed.
Total Synchronization Requests	The total number of sync requests initiated by a client.
Total Operation Replays	The total number of operation replays performed by clients on MBOs.

Messaging Package Statistics

Messaging package statistics consist of key performance indicators (KPIs) that reflect the overall function of the application environment at the cluster or domain level. The statistics highlight key totals and identify average, minimum, and maximum values for primary activities.

Note: These KPIs are not applicable at the MBO level:

- Total Subscription Commands
 - Total Devices
-

These key indicators monitor messaging packages:

KPI	Description
Total Subscription Commands	The total number of subscription commands sent from clients to the server.
Total Devices	The total number of devices involved in synchronization. This total includes the same user multiple times if he or she has multiple devices. The value comprises individual devices, and does not count multiple synchronizations requested by the same device.
Average/Minimum/Maximum Message Processing Time	The average, minimum, or maximum amount of time Unwired Server took to respond to a synchronization request message.
Time at Minimum/Maximum Processing Time	The time at which the shortest or longest response time completed.
Total Inbound Messages	The total number of messages sent from clients to Unwired Server.
Total Outbound Messages	The total number of messages sent from Unwired Server to clients.
Total Operation Replays	The total number of operation replays performed by clients on mobile business objects (MBOs).
Total Errors	The total number of errors that took place across all synchronizations.
Total Data Push	The total amount of data transmitted from the server to clients.

User Statistics

User statistics consist of key performance indicators that reflect the overall activity of application users.

User statistics can be filtered to include users who belong to a particular security configuration. This type of monitoring highlights key totals and identifies average, minimum, and maximum values for primary user activities. These calculations are dynamic, and are based on the data currently available in monitoring database for the specified time period.

Note: These statistics are not supported for Sybase Mobile CRM and Sybase Mobile Workflow for SAP application users.

Replication User Statistics

Replication user statistics reflect the synchronization activity of a group of replication-based synchronization users belonging to a specified security configuration. These statistics include general activity-related information on a per-user basis.

These key indicators monitor replication users:

KPI	Description
Total Synchronization Requests	The total number of sync requests initiated by a client.
Total Rows Received	The total number of rows received during package synchronization.
Total Rows Sent	The total number of rows sent during package synchronization.
Total Bytes Received	The total number of bytes uploaded from clients to Unwired Server.
Total Bytes Sent	The total number of bytes downloaded by clients from Unwired Server.
Average/Minimum/Maximum Synchronization Time	The average, minimum, or maximum amount of time Unwired Server took to complete a synchronization request.
Time at Maximum/Minimum Synchronization Time	The time at which the fastest or slowest synchronization is completed.
Total Operation Replays	The total number of operation replays performed by user of mobile business objects (MBOs).
Total Errors	The total number of errors that took place across all synchronizations.
Total Devices	The total number of devices involved in synchronization. This total includes the same user multiple times if he or she has multiple devices. The value comprises individual devices, and does not count multiple synchronizations requested by the same device.

Messaging User Statistics

Messaging user statistics reflect the synchronization activity of a group of messaging-based synchronization users belonging to a specified security configuration. These statistics include general activity-related information on a per-user basis.

These key indicators monitor messaging users:

KPI	Description
Total Devices	The total number of devices involved in synchronization. This total includes the same user multiple times if he or she has multiple devices. The value comprises individual devices, and does not count multiple synchronizations requested by the same device.
Average/Minimum/Maximum Message Processing Time	The average, minimum, or maximum amount of time Unwired Server took to respond to a sync request message.

KPI	Description
Time at Minimum/Maximum Message Processing Time	The time of day at which the shortest or longest message processing event completed.
Total Inbound Messages	The total number of messages sent from clients to Unwired Server.
Total Outbound Messages	The total number of messages sent from Unwired Server to clients.
Total Operation Replays	The total number of operation replays performed by clients on mobile business objects (MBOs).
Total Errors	The total number of errors that took place across all synchronizations.
Total Subscription Commands	The total number of subscription commands sent from clients to the server.
Total Data Push	The total number of import data messages.

Cache Statistics

Cache statistics provide a granular view of cache activity either at the domain or package level, particularly in the areas of cache performance, mobile business object (MBO) status, and cache group status.

Cache statistics report on performance at the domain, package, MBO, and cache group levels to allow administrators to obtain different information according to the level of specificity required. These calculations are dynamic, and are based on the data currently available in monitoring database for the specified time period.

Note: These statistics are not supported for Sybase Mobile CRM and Sybase Mobile Workflow for SAP application users.

Viewing Package-Level Cache Statistics

Use the package tree to view cache statistics at the package, cache group, or mobile business object (MBO) level.

The package tree allows for a granular view of data in all cache statistic categories except for domain-level data. Domain-level data instead uses the Filter by Domain search functionality.

1. In the left navigation pane, select **Monitoring**.
2. In the right administration pane, select **Cache Statistics**.
3. From the cache feature drop-down list, select one of the following, depending on the type of data and level of granularity you require:
 - **Domain level**
 - **Package level**
 - **Package level cache group**

- **Package level MBO**

4. Select **Show Package Tree**.

The tree view appears on the left side of the right administration pane.

5. In the tree view, click the package, cache group, or MBO for which you want to view monitoring data.

The monitoring data displays in the monitoring console. You can further filter data by specifying a time period in the search panel (for package-level cache performance and package-level MBO status only).

Cache Performance Statistics

Cache performance statistics report on key totals and identify average, minimum, and maximum values for primary cache activities. View cache performance data at the domain or package level.

Select either **Domain level** or **Package level** to view the following key performance indicators:

Key performance indicator	Description
Domain	The domain to which the package affected by the cache activity belongs.
Package	The name of the package associated with this cache activity.
Minimum/Maximum Cache Misses	The minimum or maximum number cache misses and the MBO name for which it was generated.
Minimum/Maximum Cache Hits	The minimum or maximum number of scheduled cache queries for all of the MBOs in the package in the specified date range.
Minimum/Maximum/Average % Cache Hits	The minimum or maximum percentage of scheduled cache queries for the supplied date range and the MBO name for which it was generated.
Minimum/Maximum Average Wait Time	The minimum or maximum average wait time for a scheduled cache query and the MBO name for which it was generated.
Minimum/Maximum Average Refresh Time	(Package-level only)The minimum or maximum average refresh time for an on-demand or scheduled refresh.

MBO Statistics

Mobile business object (MBO) status monitoring reports on cache activity at the MBO level, and thus, reflects activity for single mobile business objects.

Select **Package level MBO** to view the following key performance indicators:

Key performance indicator	Description
Cache Group	The name of the group of MBOs associated with this cache activity.
MBO	The name of the single mobile business object associated with this cache activity.
Number of Rows	The number of rows affected by the cache refresh.
Cache Hits	The number of scheduled cache queries that occurred in the supplied date range.
Cache Misses	The number of on-demand cache or cache partition refreshes that occurred in the supplied date range.
Access Count	The number of cache queries that occurred in the supplied date range.
Minimum/Maximum/Average Wait Time	The minimum, maximum, or average duration of cache queries in the supplied date range. This time does not include the time required to refresh the cache in a cache “miss” scenario. Instead Minimum/Maximum/Average Full Refresh Time exposes this data.
Minimum/Maximum/Average Full Refresh Time	The minimum, maximum, or average duration of on-demand and scheduled full refresh activities in the supplied date range.

Cache Group Status Statistics

Cache group status statistics provide monitoring data about cache activity at the cache group level. The data reflects activity for all mobile business objects (MBOs) belonging to a cache group.

Select **Package level cache group** to view the following key performance indicators (KPIs):

KPI	Description
Package	The name of the package to which the associated cache group belongs
Cache Group	The name of the group of MBOs associated with the cache activity
Number of Rows	The number of rows in the cache table of the MBO
Last Full Refresh Time	The last time the cache or cache partition was fully refreshed
Last Update Time	The last time a row in the cache was updated for any reason (row-level refresh, full refresh, partitioned refresh, alternate read, or data change notification)
Last Invalidate Time	The last time the cache was invalidated

KPI	Description
Cache Coherency Window	<p>The data validity time period for the cache group, in seconds. Can span any value in this range:</p> <ul style="list-style-type: none">• 0 shows that data is always retrieved on-demand for each client.• 2049840000 shows that the cache never expires. This occurs when you set the on-demand cache group to NEVER expire or scheduled cache group to NEVER repeat.

Monitor

Troubleshoot the Sybase Control Center

Troubleshoot issues that arise in Sybase Control Center for Unwired Platform.

Using Sybase Control Center to Troubleshoot Unwired Platform

Problem: Unwired Platform is not functioning properly or exhibits abnormal behaviour.

Consult these Sybase Control Center sources to find useful information to help you troubleshoot Unwired Platform issues:

1. Review the server log – view server errors, warnings, and general information to identify problems. Access the Server node in the left navigation tree of Sybase Control Center to view server log data.
2. Review domain logs – if domain logging is enabled, view domain logs in each Domains > <DomainName>> Log node of Sybase Control Center. Aggregated log data in the console makes domain information readily accessible and actionable.
3. Review monitoring data – access the Monitoring node in the left navigation tree of Sybase Control Center to view monitoring data on the following components of Unwired Platform: replication-based synchronization, messaging-based synchronization, messaging queue, data change notifications, device notifications, packages, users, and cache. See *System Diagnostics* in *System Administration* .
4. Review Application Connection status – access the Applications node in the left navigation pane of Sybase Control Center to view application connection information in the right pane.

Note: You can also view domain-level Application Connection status – navigate to the domain then select **Applications** in the left navigation pane, and view application connection information in the right pane.

5. Review package client logs – access the Client Log tab of the Packages > <PackageName> node in Sybase Control Center to view data about client application operations for all devices subscribed to a package. This information allows you to track errors and identify performance issues.
6. Review MBO and operation history – access the History tab for both the MBO and operation nodes of a package in Sybase Control Center to review error history during synchronizations and operation replays.

Collecting Administration Performance Data for Troubleshooting

Problem: You need to collect performance data to troubleshoot performance issues in Sybase Control Center for Unwired Platform administrative options.

Solution: Set up the `<UnwiredPlatform_InstallDir>\SCC-XX\log\executionTime.log`, which provides information on the length of time taken to complete operations in Sybase Control Center. Sybase Product Support and Engineering teams can use this information to diagnose the source of your performance issues. To set up this log file:

1. Open `<UnwiredPlatform_InstallDir>\SCC-XX\plugins\com.sybase.supadminplugin\agent-plugin.xml`.
2. Add the following line to the file under the `<properties>` element:

```
<set-property property="log_MO_method_execution_time" value="enable_log_mo_method_execution_time" />
```
3. Open `<UnwiredPlatform_InstallDir>\SCC-XX\conf\log4j.properties`.
4. If you are experiencing log truncation issues, edit the following lines to change the default values for maximum file size (default: 5MB) and maximum backup index (default: 10 files) to the values shown in this example:

```
## file appender (size-based rolling)
log4j.appender.executionTime=org.apache.log4j.RollingFileAppender
log4j.appender.executionTime.File=${com.sybase.ua.home}/log/
executionTime.log
log4j.appender.executionTime.layout=org.apache.log4j.PatternLayout
log4j.appender.executionTime.layout.ConversionPattern=%d [%-5p]
[%t] %c.%M(%L) - %m%n
log4j.appender.executionTime.MaxFileSize=50MB
log4j.appender.executionTime.MaxBackupIndex=20
## log MO method execution time
log4j.logger.com.sybase.uep.sysadmin.management.aop=INFO,executionTime
```

5. Restart SCC.
The `executionTime.log` file now appears in the `<UnwiredPlatform_InstallDir>\SCC-XX\log` folder.

Use this log file to diagnose and analyze performance problems. For more information on configuring the `agent-plugin.xml` configuration file, search for *Agent Plugin Properties Reference* in the *System Administration* guide.

You can also use the Adobe Flex log to track performance in Sybase Control Center. To access Flex-side logging, highlight the resource in the Perspective Resources view and select View Log to show the user interface time for each activity. Alternately:

1. Modify the `<UnwiredPlatform_InstallDir>\SCC-XX\plugins\com.sybase.supadminplugin\agent-plugin.xml` file as indicated in step 2, above.
2. Restart SCC.
3. Log in and perform your regular administrative tasks.
4. View the execution time indicators for these operations in the cookie file `supatcookie.sol`. The location of this file varies depending on your operating system:

Operating System	Location
Windows XP	C:\Documents and Settings\ <i><username></i> \Application Data\Macromedia\Flash Player\#SharedObjects
Windows Vista	C:\Users\ <i><username></i> \AppData\Roaming\Macromedia\Flash Player\#SharedObjects
Macintosh OS X	/Users/ <i><username></i> /Library/Preferences/Macromedia/Flash Player/#SharedObjects
Linux	/home/ <i><username></i> /.macromedia/Flash_Player/#SharedObjects

5. Analyze the log using your preferred method of data analysis.

Sybase Control Center Management Tier Issues

Review this list of documented general issues for Sybase Control Center and its server management-related services.

Launching SCC Results in Rounded Rectangle Box or Empty Console Screen

Problem: When you launch Sybase Control Center, a rounded rectangular box appears instead of the administration console, or the console displays a gray or empty screen.

Explanation: The Adobe Flash Player version is older than the minimum version supported by SCC.

Troubleshoot the Sybase Control Center

Solution: Upgrade your Flash Player version to the latest version. For more information on software prerequisites, see *Supported Hardware and Software*.

Sybase Control Center Windows Service Fails to Start

Problem: When starting the Sybase Control Center *X.X*service, it takes a long time before failing, and the service manager displays a message that the service startup has timed out.

The <UnwiredPlatform_InstallDir>\SCC_X-X\log\agent.log shows the following message:

Explanation: This problem usually occurs when the Sybase Control Center repository database log file is out of sync with the repository database. A related symptom is the message `SQL Login Failure` in the Sybase Control Center repository log file.

Solution 1: Review <UnwiredPlatform_InstallDir>\SCC-X_X\services\Repository\scc_repository.log log for any issues with the database transaction log file during startup. If the transaction log could not be processed, the database cannot start, and consequently nor can the Sybase Control Center service. Resolve this error by:

1. Creating a backup of <UnwiredPlatform_InstallDir>\SCC-X_X\services\Repository\scc_repository.log.
2. Deleting the <UnwiredPlatform_InstallDir>\SCC-X_X\services\Repository\scc_repository.log file and restarting the Sybase Control Center service.

Solution 2: Review <UnwiredPlatform_InstallDir>\SCC-X_X\services\Repository\scc_repository.log log for any failures in database transaction and/or recovery. Resolve this error by temporarily configuring the repository database (-f) to start without a transaction log:

1. Log out of Sybase Control Center and then shutdown Sybase Control Center service.
2. Open command prompt window, and run the following command:

```
C:\Sybase\SCC-3_2\services\SccSADatasever\sa
\bin_windows32\dbsrv11.exe -n scc_repository -o C:\Sybase
\SCC-3_2\services\Repository\scc_repository.slg -f -m -qi -
qw -sb 0 -gn 100 -gm 500 -zl -zp -x TCPIP{port=3638} C:
\Sybase\SCC-3_2\services\Repository\scc_repository.db
```

3. Delete the <UnwiredPlatform_InstallDir>\SCC-X_X\services\Repository\scc_repository.log file using Windows Explorer.
4. Restart the Sybase Control Center service.

Sybase Control Center Windows Service Deleted

Problem: the Sybase Control Center *X.X* windows service was inadvertently deleted, so Sybase Control Center is unavailable.

Solution: Re-create the Windows service with the following command:

```
UnwiredPlatform_InstallDir\SCC-X_X\utility\ntautostart\release
\sccservice.exe -install
```

Sybase Control Center Fails to Start

Problem: The Sybase Control Center server does not start.

This problem occurs when the host name cannot be resolved or the IP address of the machine has changed since the product installation. This troubleshooting topic applies only when either of these scenarios is true.

Solution 1: Change the host name to its IP address in the Sybase Control Center `service-config.xml` file:

1. From the command line, verify the host name by running `nslookup<hostname >`.
2. If the DNS server cannot resolve the host name, edit the colocated `<UnwiredPlatform_InstallDir>\SCC-XX\services\RMI\service-config.xml` file:
 - a. Log out of Sybase Control Center.
 - b. Stop the Sybase Control Center *X.X* service.
 - c. Open `<UnwiredPlatform_InstallDir>\SCC-XX\services\RMI\service-config.xml`.
 - d. Locate this line: `<set-property property="address" value="<hostname>" />`.
If the line does not exist, add it under the `<properties></properties>` element in the file.
 - e. Change the value from the host name to the IP address of the host computer. If the IP address is already used, ensure it is valid (especially if the IP address has recently been changed).
 - f. Restart the Sybase Control Center *X.X* service.
 - g. Log in to Sybase Control Center and proceed with your administrative tasks.

Second Sybase Control Center Fails to Start

Problem: Cannot start a second co-existing Sybase Control Center in a deployment environment.

Explanation: When multiple versions of Sybase Control Center co-exist on a single machine, if the older version is already using the default port number, the new version of Sybase Control Center uses another port number, such as 8285. If the configuration files have not been updated, this may cause port conflicts.

Troubleshoot the Sybase Control Center

Solution: Check the port numbers, and check the configuration files to make sure the configuration is correct. See the topic *Port Number Reference* in the *System Administration* guide. If the configuration is correct, you may need to start the second version of Sybase Control Center manually.

Login Invalid in Sybase Control Center

Problem: Logging in to Sybase Control Center generates an Invalid Login message.

Solution:

- Verify Sybase Control Center session validity – ensure that the current Sybase Control Center session is active. If the session is frozen or expired, refresh the page or close the browser and try again.
- Verify authentication configuration – ensure that the Sybase Control Center authentication provider configuration is correct, and points to the correct server. See *Enabling Authentication and RBAC for Administration Logins* in the *Security* guide.
- Verify LDAP consistency – if you are using LDAP security – ensure that the LDAP supports administration roles used by Unwired Platform.
- Check `<UnwiredPlatform_InstallDir>\SCC_XX\log\agent.log` for any issues with starting various services.
- If all services are running, check the `<UnwiredPlatform_InstallDir>\SCC_XX\log\agent.log` for an error message containing text similar to the following:
Failed to authenticate user 'supAdmin' (Failed to connect to service:jmx:rmi:///jndi/rmi://eas3w03.sybase.com:9999/agent, probably because the agent is protected and requires credentials.Security Service Error. Agent service exception.)
 - Ensure that the Sybase Control Center authentication provider configuration is correct, and points to the correct server. See *Enabling Authentication and RBAC for Administration Logins* in the *Security* guide.

Login Fails in Sybase Control Center

Problem: Removed the PreconfiguredUser login module from the "admin" security configuration. Now, logins to Sybase Control Center fail.

YSolution: Verify server-side configuration by trying to connect to Unwired Server from Unwired WorkSpace (requires creating Unwired Server Connection Profile with proper username and password among other things). If that works but the SCC login still fails, check for error messages in `inagent.log` can offer further clues on the

Administrator Account is Locked

Problem: An administrator tried logging into Unwired Server from Sybase Control Center multiple times. After receiving multiple Wrong username and password errors,

finally a The account is currently locked. Please contact your server administrator. message is displayed.

Solution: The platform administrator uses to properties to control the number of login attempts that cannot be exceeded: the login failur account lock threshold and and timeout. When a user passes this threshold value with failed login attempts the account is locked. A user must wait for the lock timeout value to pass and can try logging in again.

For details, see *Creating a Security Configuration* in Sybase Control Center online help.

Browser Refresh (F5) Causes Logout

Problem: Pressing the **F5** key to refresh your browser logs you out of Sybase Control Center.

Solution: Do not use **F5** when you are logged in to Sybase Control Center. Browser refresh does not refresh data inside Sybase Control Center, but refreshes the loaded application or pages in the browser—in this case, the Adobe Flash on which Sybase Control Center is built. Consequently, pressing **F5** logs you out of any servers you are currently logged in to, including Sybase Control Center.

Stale Version of Sybase Control Center After Upgrade

Problem: after upgrading Sybase Unwired Platform and relaunching Sybase Control Center through a Web browser, a stale version of Sybase Control Center loads in the browser.

Explanation: Adobe® Flash® Player caches the earlier version of Sybase Control Center locally, preventing you from logging in to the correct version of Sybase Control Center when accessing the browser.

Solution 1: Clear the Adobe Flash Player cache:

1. In Windows Explorer, navigate to `C:\Documents and Settings\\Application Data\Macromedia\FIash Player\#SharedObjects`, and delete all files in this folder.

As an alternative to manually deleting files, you can also access the Adobe Flash Player Cache Cleanup URL: http://www.macromedia.com/support/documentation/en/flashplayer/help/settings_manager07.html

Solution 2: Only perform this solution if Solution 1 does not solve the problem. Clear browser history:

1. In Microsoft Internet Explorer, select **Tools > Internet Options > General > Delete...** and delete all temporary files, history, cookies, saved passwords, and Web form information.

Sybase Control Center Reports Certificate Problem

When attempting to bring up Sybase Control Center by clicking the SCC link after installation, this message appears: There is a problem with this website's security certificate.

Explanation: This can occur when the browser session starts on the same computer as Sybase Control Center. The installer automatically sets up a local security certificate, but the

Troubleshoot the Sybase Control Center

certificate installed for HTTPS in the web container keystore is a self-signed root certificate, which is not recognized by the client browser.

Solution: Follow browser-specific instructions to accept the certificate into the Windows certificate store. Once the certificate is accepted, you may also need to change the SCC Web URL to include the network domain name <yourco.com> in addition to the host name. That host name in the Web URL must match with the "Issued To" property of the certificate.

Previous Administrator Credentials Used

Problem: You cannot use new credentials to authenticate against a resource in Sybase Control Center. When an administrator enters credentials with the **Remember these credentials for future sessions** option, Sybase Control Center uses those credentials until they are cleared.

Solution: Clear credentials so that Sybase Control Center does not use them for future sessions:

1. Open the Perspective Resources window.
2. Select the resource you want to log in to.
3. From the menu bar, select **Resource > Clear Authentication Parameters** and click **OK**.

You can now authenticate against the resource using new administrator credentials.

Security Error Triggered When Connecting to SCC from Remote Browser

Problem: Connecting to Sybase Control Center from a browser that is remote triggers a security exception.

Solution: Ensure you have a security certificate installed in the Windows security store. See *Setting Up Browser Certificates for Sybase Control Center Connections* in Sybase Control Center online help.

Administrator Login Passes When Provider Is Not Available

Problem: The configured authentication provider is unavailable but administration credentials are still accepted.

Explanation: The administrator login credentials may be cached by Unwired Server.

Solution: If this behavior is undesired, reduce the cache timeout value used by the Unwired Server security domain instance. For details, search for *Authentication Cache Timeouts* in the *Security* guide.

Host Name of Registered Resource Changed But Is Not Updated

Problem: An administrator changes the host name property of a registered resource; but in Sybase Control Center, the old host name is still used and the management console for Unwired Platform does not appear.

Description: If you modify the resource properties for an Unwired Server in Sybase Control Center, the new host name or IP address is not used in establishing a connection to the server.

Solution: After changing the host name property of the resource, in the Perspective Resources view, right-click the resource and select **Authenticate** to update resource connection properties. You can then launch the management console successfully.

Management Issues with Clustered Data Tiers

Problem: if you install Unwired Platform and the cache database on Microsoft Cluster, you will receive errors when trying to manage the cluster in Sybase Control Center. This is because Microsoft Cluster uses node switches.

Solution: Replace the current entry for the cluster with a new entry that uses the computer node's hostname or IP address, rather than Unwired Platform cluster's hostname (the default).

Poor Sybase Control Center Performance after Upgrade

Problem: After upgrading to the latest version of Sybase Unwired Platform, Sybase Control Center performance is poor.

Explanation: This may indicate that Flash Player cache from the previous version of Sybase Control Center is filled and slowing down performance.

1. Navigate to C:\Documents and Settings\username\Application Data\Macromedia\Flash Player\#SharedObjects.
2. Delete all files under this folder.

Note: Alternatively, go to the following link from a browser: http://www.macromedia.com/support/documentation/en/flashplayer/help/settings_manager07.html. Use the Website Storage settings panel to change storage capacity, or delete Websites to clean up the cache.

Sybase Control Center Communication with Unwired Server Fails

Problem: While using Sybase Control Center, a Communication with Unwired Server failed appears.

Explanation: Sybase Control Center cannot connect to the Unwired Server and displays this error message. To confirm this issue, open the Sybase Control Center <SCC_HOME>\log\gateway.log and look for org.omg.CORBA.COMM_FAILURE, com.sybase.djc.rmi.iiop.BadMagicException, or org.omg.CORBA.MARSHAL entries.

Troubleshoot the Sybase Control Center

Solutions:

1. Ensure the protocol and port used by both the Unwired Server management port and the Sybase Control Center managed resource registration entry match. For information about validating and changing these properties, see *Adding or Updating Unwired Server Registration Properties* in the Sybase Control Center online help.
2. Validate that the configured port is the Unwired Server management port of 2000 or 2001. Sybase recommends that you not change these default values. If you have changed them and the connection fails, update the managed resource connection property to use the default.

For more information about ports, see Port Number Reference in *System Administration*. For more information about validating and changing this port, see *Registering a Resource as an SCC Managed Resource* in the Sybase Control Center online help.

3. If the management security profile now uses SSL mutual authentication (`mutual_auth`), validate that you have installed certificates for mutual authentication into both Sybase Control Center's and Unwired Server's keystore. If each component doesn't have the opposite set of certificates, mutual authentication fails. Either install the missing certificates if mutual authentication is required, or use the following procedure to recover from this scenario:
 1. If Unwired Server also has a standard management (non-secure) port available, you can connect to that port by updating the Sybase Control Center resource (localhost) port number property and setting `secure` to "No". See *Adding or Updating Unwired Server Registration Properties* in Sybase Control Center online help.
 2. If Unwired Server doesn't have the standard management port enabled, then update the **securityProfile** property value in the `<UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\UnwiredServer\Repository\Instance\com\sybase\djc\server\SocketListener\{ServerName}_iiops1.properties` file to use the "default" profile, and restart Unwired Server.

For information about installing certificates, see *Changing Installed Certificates Used for Unwired Server and Sybase Control Center HTTPS Listeners*, in the *Security* guide. For information about changing the authentication method used by the security profile, see *Creating an SSL Security Profile in Sybase Control Center* in Sybase Control Center online help.

Platform Component Monitoring Issues

Review this list of documented issues for platform components monitored by Sybase Control Center.

Monitoring Data Does Not Appear in History Tab

Problem: Monitoring data does not appear immediately in the History tab.

Explanation: The monitoring data is stored in memory to optimize database access, and periodically flushed to the monitoring database.

Solution: Try either of these options:

- Wait for the data to be flushed. The default time period is five minutes.
- Change the flush interval to a smaller value in Sybase Control Center:
 1. In the left navigation pane, select **Monitoring**.
 2. In the right administration pane, select the **General** tab.
 3. Click **Configuration**.
 4. In the flush threshold section, ensure that **Enable flush threshold configuration** is selected.
 5. Select one of:
 - **Number of rows** – monitoring data that surpasses the specified number of rows is flushed from the console display. Enter the desired number of rows adjacent to **Rows**. The default is 100.
 - **Time interval** – monitoring data older than the specified time interval is flushed from the console display. Enter the desired duration adjacent to **Minutes**. The default is 5.
 - **Either rows or time interval** – monitoring data is flushed from the console display according to whichever value is reached first: either the specified number of rows or the specified time interval. Enter the desired rows and duration adjacent to **Rows** and **Minutes**, respectively.
 6. Retrieve the results list using the Sybase Control Center monitoring node.

Domain Log Data Does Not Appear in History Tab

Problem: Domain log data does not appear immediately in the History tab.

Explanation: The domain log data is stored in memory to optimize database access, and periodically flushed to the domain log database.

Solution: Try either of these options:

- Wait for the data to be flushed. The default time period is five minutes.
- Change the flush interval to a smaller value in Sybase Control Center:
 1. In the left navigation pane, expand the **Domains** folder and select the default domain.
 2. Select **Log**.
 3. In the right administration pane, select the **Settings** tab.
 4. Click **Configuration**.

5. In the flush threshold section, ensure that **Enable flush threshold configuration** is selected.
6. Select one of:
 - **Number of rows** – domain log data that surpasses the specified number of rows is flushed from the console display. Enter the desired number of rows adjacent to **Rows**. The default is 100.
 - **Time interval** – domain log data older than the specified time interval is flushed from the console display. Enter the desired duration adjacent to **Minutes**. The default is 5.
 - **Either rows or time interval** – domain log data is flushed from the console display according to whichever value is reached first: either the specified number of rows or the specified time interval. Enter the desired rows and duration adjacent to **Rows** and **Minutes**, respectively.
7. Retrieve the results list using the Sybase Control Center domain log node.

Previously Existing Monitoring Data No Longer Appears

Problem: Monitoring data that displayed previously no longer appears.

Explanation: By default, monitoring data is preserved in the database for seven days. After that period, the data is removed.

Solution: Change the auto purge setting value in Sybase Control Center. Auto purge clears obsolete data from the monitoring database once it reaches the specified threshold.

1. In the left navigation pane, select **Monitoring**.
2. In the right administration pane, select the **General** tab.
3. Click **Configuration**.
4. In the auto purge section, ensure that **Enable auto purge configuration** is selected.
5. Enter the length of time (in days) to retain monitoring data before it is purged.
6. Restart the server.
7. Retrieve the results list using the Sybase Control Center monitoring node.

Previously Existing Domain Log Data No Longer Appears

Problem: Domain log data that displayed previously no longer appears.

Explanation: By default, domain log data is preserved in the database for seven days. After that period, the data is removed.

Solution: Change the auto purge setting value in Sybase Control Center. Auto purge clears obsolete data from the domain log database once it reaches the specified threshold.

1. In the left navigation pane, expand the **Domains** folder and select the default domain.
2. Select **Log**.
3. In the right administration pane, select the **Settings** tab.

4. Click **Configuration**.
5. In the auto purge section, ensure that **Enable auto purge configuration** is selected.
6. Enter the length of time (in days) to retain domain log data before it is purged.
7. Restart the server.
8. Retrieve the results list using the Sybase Control Center domain log node.

Server Tier Administration Issues

Review this list of documented issues for Unwired Server or its internal synchronization services configured and administered by Sybase Control Center.

Server List Not Retrieved

Problem: No list of Unwired Servers displays in Sybase Control Center. Instead, an `Error Retrieving Server List` message appears in the left navigation pane.

Scenario 1: No other error message appears.

If this is the case, one of the following explanations may apply:

- You are attempting to connect to a remote server that is not properly registered in Sybase Control Center.

Solution: Manually register the remote server. By default, only Unwired Servers installed to the same host computer are automatically registered with Sybase Control Center. See *Getting Started with Unwired Server Administration* in the Sybase Control Center online help. If you have recently made changes to the environment, for example, by modifying server resource properties (login, password, host name, IP address, or port number), ensure that you reauthenticate after making the changes.

- Jetty caching in Sybase Control Center prevents the console from displaying the server tree. This is indicated by 404 errors in both the console URL and `<UnwiredPlatform_InstallDir>\SCC-X_X\services\EmbeddedWebContainer\log\http-service.log` (the HTTP access log).

Solution:

1. Close Sybase Control Center.
2. Stop Sybase Control Center X.X Service.
3. Delete the contents of: `<UnwiredPlatform_InstallDir>\SCC-X_X\services\EmbeddedWebContainer\container\Jetty-X.X.XX\work`.
4. Restart Sybase Control Center X.X service.

Scenario 2: The right administration pane shows an `Authentication has failed` error message.

If this is the case, one of the following explanations may apply:

Troubleshoot the Sybase Control Center

- You have not performed the "Authenticate" step in Sybase Control Center after registering the resource or changing their credentials.

Solution: In the Perspective Resources view, right click the server name and select **Authenticate**. In the default configuration, if you have used "supAdmin" to log in to Sybase Control Center, select **Use my current SCC login**.

- The server IP may have changed.

Solution: Update server resource properties, and repeat the "Authenticate" step described above. See the topic *Sybase Control Center Fails to Start*.

Scenario 3: The right administration pane shows a Connection unknown. Ensure Server is running... message.

If this is the case, one of the following explanations may apply:

- Unwired Server responded with an exception indicating a problem on the server.

Solution: Check `<UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\UnwiredServer\logs\<hostname>-server.log` for details.

- The Sybase Control Center security provider is down or a system condition prevents Sybase Control Center from authenticating the user for administration access.

Solution: Ensure that the security provider is running and that its host is reachable from the Sybase Control Center host.

Scenario 4: In some rare cases, the connection between Sybase Control Center and Unwired Server cannot be established after trying the previous recommendations.

Solution: You may need to stop and restart the Sybase Control Center *XX* windows service. After stopping the window service, make sure the process `uaservices.exe` is not running (or stop it from Windows task manager). Then log in to Sybase Control Center again.

Scenario 5: This may happen if you upgraded Sybase Unwired Platform to a newer version, and changed the server host name.

Solution: You need to complete some extra steps:

1. Change the listener prefix of `httpListeners` and `iiopListeners` for the new hostname in the new server's properties file:

```
Repository\Instance\com\sybase\djc\server\ApplicationServer
\default.properties, <new_hostname>.properties
```

2. In `Repository\Instance\com\sybase\djc\server\SocketListner*.properties`, rename all the `<old_hostname>_<protocol>.properties` into `<new_hostname>_<protocol>.properties`.

3. Use `dbisqlc` to update the table: `cluster_installation` in `clusterdb`, `update cluster_installation set hostname='<new_hostname>' where hostname='<old_hostname>'`.

Unwired Server Fails to Start

Problem: Starting Unwired Server from Windows services or the desktop shortcut fails.

Solution:

1. Ensure that the server license is valid and has not expired.
2. Open Windows services to check that the services Unwired Server depends on for start-up are running properly. Identify dependencies by right-clicking the service and selecting **Properties**.
3. Check `<UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\UnwiredServer\log\<serverName>-server.log` for error messages indicating the nature of Unwired Server start-up issues.
4. Check `<UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\UnwiredServer\log\bootstrap**.log` for possible license errors.

Error in Listing Application Connections and ADMIN_WEBSERVICE_INVOCATION_ERROR in gateway.log

Problem: This message may indicate that an Unwired Server administrative component is not running.

If users report a problem listing application connections in Sybase Control Center, check for this error message in the Sybase Control Center gateway.log file:

```
com.sybase.uep.sysadmin.management.mbean.UEPAdminException:
com.sybase.uep.admin.client.AdminException:
ADMIN_WEBSERVICE_INVOCATION_ERROR: java.security.PrivilegedActionExc
eption: com.sun.xml.internal.messaging.saaj.SOAPEXceptionImpl:
Message send failed
javax.management.MBeanException:
```

Explanation: Usually this occurs when when there is a conflict on the currently configured port for the administration web service or a component of Sybase Unwired Server service went down for some reason.

One way to verify availability of the Web service is by accessing the following URL from the host where Sybase Unwired Platform is installed: `http://localhost:5100/MobileOffice/Admin.asmx`. The default Messaging port is 5100, but this may vary depending on your configuration.

Solution 1: Check Windows Application Event log for any error reported there. If the service is configured to run with a domain account and the password has been changed, you will need to update the password.

Solution 2: Make sure the administration Web service is up and running, and correctly configured. Review *Cannot Access Applications Tab and Web Service Error* in *Troubleshooting* to reconfigure the port in case of conflict with existing port.

Starting or Restarting a Remote Server from Sybase Control Center Fails

Problem: After you have registered a remote server in Sybase Control Center, you cannot start or restart the server.

If the DNS server cannot resolve the host name of the machine on which the remote Unwired Server is installed, or if the host has no internal DNS server, you cannot start, stop, or restart that Unwired Server using your local instance of Sybase Control Center. Because this network communication relies on name resolution, you must ensure that DNS is set up properly to successfully control a remote Unwired Server.

Before attempting the following solutions, verify that:

1. Sybase Control Center is running on the remote host.
2. A network connection can be established between your Sybase Control Center host and the Sybase Control Center agent on the remote server's host.

If the DNS server cannot establish a connection, try the following:

Solution 1: Repair the network DNS server setup. If you or your network administrator cannot modify the DNS, use solution 2.

Solution 2: Change the host name to its IP address in the Sybase Control Center `service-config.xml` file:

- If you cannot resolve the local host name, modify the file on the local instance of Sybase Control Center.
 - If you cannot resolve the remote host name, modify the file on the remote instance of Sybase Control Center.
 - If you cannot resolve both the remote and local host names, modify both files.
1. From the command line, verify the host name by running `nslookup<hostname>`.
 2. If the DNS server cannot resolve the host name, edit the colocated `<UnwiredPlatform_InstallDir>\SCC-XX\services\RMI\service-config.xml` file:
 - a. Log out of Sybase Control Center.
 - b. Stop the Sybase Control Center X.X service.
 - c. Open `<UnwiredPlatform_InstallDir>\SCC-XX\services\RMI\service-config.xml`.
 - d. Locate this line: `<set-property property="address" value="<hostname>" />`.
If the line does not exist, add it under the `<properties></properties>` element in the file.

- e. Change the value from the host name to the IP address of the host computer. If the IP address is already used, ensure it is valid (especially if the IP address has recently been changed).
- f. Restart the Sybase Control Center X.X service.
- g. Log in to Sybase Control Center and proceed with your administrative tasks.

If the DNS server resolves the host name, but the problem persists, check that both:

- The remote host on which Unwired Platform and Sybase Control Center are installed can receive UDP multicasts from the local host on which Sybase Control Center is installed, and
- The remote instance of Sybase Control Center uses RMI port 9999.

Solution 3: Make sure the `hosts` file includes complete entries for each node in the Unwired Server cluster.

1. On each Unwired Server host, edit the `hosts` file, located at:
C:\WINDOWS\system32\drivers\etc
2. Add entries to identify the IP address and fully qualified network name of every other node in the Unwired Server cluster.

Port Conflict Issues

Problem: You have identified a Sybase Control Center X.X service port conflict.

Solution:

1. Identify the service with the port conflict in <UnwiredPlatform_InstallDir>\SCC-X_X\log\agent.log.
2. Use a text editor to open <UnwiredPlatform_InstallDir>\SCC-X_X\Services\<Servicename>\service-config.xml.
3. Change the port to an available port number.
4. Save and close the file.

Search for *Port Number Reference* in *System Administration* for more information.

Unexpected Listener Startup or Connection Errors

Problem: You encounter unexpected listener startup or connection errors for Unwired Platform components. This is usually seen when Sybase Unwired Server is installed on a host in DMZ (De-Militarized Zone) within the internal and external firewalls.

Solution:

1. Verify that the TCP/IP filtering restriction is not in effect on the host machine.

To do so on Windows XP, navigate to: **Control Panel > Network Connections > Local Area Connection 1 > Properties > General tab > Internet Protocol (TCP/IP) > Properties > General tab > Advanced > Options tab > TCP/IP filtering > Properties**

2. In TCP/IP Filtering, check to make sure the Enable TCP/IP Filtering (All Adopters) checkbox is not selected. This enables all Sybase Unwired Platform infrastructure ports. If you do choose to select it, be sure to select Permit All for TCP Ports to enable all Sybase Unwired Platform infrastructure ports. These ports are documented in the Installation Guide.
3. Click **OK** to close each window and save your changes.
4. You can change “Local Area Connection 1” to the network connection name being used on the machine.
5. Make sure users are not using third party port blockers, like McAfee Antivirus.

Refreshing Server Configuration Displays Only Partial Updates

Problem: The Refresh button in the Server Configuration node does not display correct properties or values, despite changes being made and saved. Updates consequently appear to have been lost. In some scenarios, when you save the Server Configuration, it fails with the message `Save Failed`.

Scenario 1: After restarting Unwired Server, refreshing the server configuration displays the first saved change, but not subsequent saved updates. The message `Save Failed` appears in the administration console after you attempt to save an update.

In this scenario, the second save was likely unsuccessful. The message `Save Failed` indicates a conflict with the first set of updates.

Cumulative saved changes are applied successfully upon server restart only if these updates do not conflict. Attempting to save two conflicting sets of changes fails.

Solution: Inject a server restart in between each saved change to ensure that the required updates are propagated across the server.

Scenario 2: After restarting Unwired Server, refreshing the server configuration displays the final saved update, but not previous ones.

The refresh action following saved configuration changes must be used in conjunction with an Unwired Server restart. Refreshing the server configuration displays the latest successfully saved configuration information.

If you click Refresh in between two sets of saved changes, only the most recent saved updates are applied during a server restart, as in the following workflow:

1. Make the first change.
2. Save the configuration.
3. Refresh the configuration.
4. Make the second change.

5. Save the configuration.
6. Restart the server.
7. Refresh the configuration.

In this sequence, only the second set of changes in step 4 are committed and consequently displayed as the current set of properties used by Unwired Server.

Solution: If you refresh the configuration after saving updates to it, restart Unwired Server immediately to apply those changes before making another set of updates. Otherwise, the first set of configuration changes will be lost. The Refresh button allows you to then validate that those changes are applied and used by Unwired Server. For details on how to refresh the server in the correct sequence, see *Saving and Refreshing an Unwired Server Configuration* in the Sybase Control Center online help.

Users Connect with Old Credentials

Problem: A user changes password in the backend security system, but can still authenticate with the previous password when connecting to Unwired Server.

Description: Unwired Server securely caches authenticated login credentials (1 hour by default), so that subsequent connection requests using the same credentials are not sent to the underlying security provider until the login cache timeout is reached. However, if the same user uses changed credentials, the authentication request is sent to the underlying security provider. The authorization outcome is not cached and always delegated to the security provider in the security configuration.

Solution: To reduce the cache period, decrease the default authentication cache timeout for a security configuration using Sybase Control Center (go to the Cluster > Security > <security configurationname> > Settings tab). Setting the property to 0 results in disabling the authentication caching (not recommended for performance reasons).

AuthorizationException Displays Instead of Status

The SCC administration console left-pane tree structure is not complete, and an AuthorizationException is reported..

Explanation: This may happen if the SCC administration console internal network communications are not working properly.

Solution:

1. Close the Internet Explorer session.
2. Relaunch the SCC administrative console.
3. Log in as usual.

The internal network connection is resumed by restarting, so the tree displays information and status properly.

Increasing Messaging Queue Counts Degrades Performance

Problem: Both inbound and outbound messaging queue counts were increased, however, performance degraded as a result.

Description: After increasing inbound and outbound message queue count, the default maxThreads of IIOP socket listener is insufficient.

Solution: Increase the maxThreads of IIOP socket listener by editing the `<hostname>_iiopl.properties` file (located in `<UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\UnwiredServer\Repository\Instance\com\sybase\djc\server\SocketListener\`), and restart Unwired Server. The maxThread of IIOP socket listener must be larger than the sum of all nodes needed IIOP thread counts.

Saving Server Configuration Fails Due to Certificate Validation Error

Problem: Saving the server configuration after property updates yields this error:
"[com.sybase.sup.admin.server.configuration.RuntimeServerConfigurationHandler] Invalid configuration object for: SyncServerConfiguration. Message : 'certificate validation failed. Update did not happen.'"

Solution: The message suggests that the server certificate has expired. Update the certificate file to a non-expired version, and try to save again.

Unknown Server Error Message

Problem: An internal server error occurs.

Solution: Check the logs for more details. The following two log files may provide more information: `\SCC-X_X\log\gateway.log` and `\UnwiredPlatform\Servers\UnwiredServer\logs\{ServerName}-server.log`.

Package Deployment and Management Issues

Review this list of documented issues for packages deployed or managed from Sybase Control Center.

Exporting or Deploying Large Packages Fails

Problem: You used Sybase Control Center to export or deploy a large package, and it fails.

You can troubleshoot this error by opening the Sybase Control Center `<UnwiredPlatform_InstallDir>\SCCXX\log\agent.log` file and checking for a message that is similar to this one:

```
exception: java.lang.IllegalStateException: Form too large
```

Explanation: This message means that the package, and not the form, is too large. The Web server that hosts Sybase Control Center cannot manage the data. A number like 273310 indicates the size of the package in kilobytes (that is, 273,310).

Solution 1: Use this solution if you run the Sybase Control Center *XX* as a service (default).

1. Open `<UnwiredPlatform_InstallDir>\SCC-X_X\bin\scc.properties` in a text editor
2. Set `MAX_FORM_CONTENT_SIZE` and `jvmopt1`. For example:

```
jvmopt=-Dorg.mortbay.jetty.Request.maxFormContentSize=2000000
```
3. Save the file.

Solution 2: Use this solution if you do not run the Sybase Control Center *XX* as a service.

1. Close Sybase Control Center, and stop Sybase Control Center *XX* using the Windows Services dialog.
2. Open `<UnwiredPlatform_InstallDir>\SCC_X-X\bin\scc.bat`, in a text editor.
3. Set the `maxFormContentSize` property to a value larger than the default, and save the file. The default is 2000000. For example:

```
-Dorg.mortbay.jetty.Request.maxFormContentSize=2048000
```
4. Restart the Sybase Control Center *XX* (using the updated `scc.bat` file), and reopen Sybase Control Center.

Invalid DOE-C User Error for an SAP Server Connection

Problem: The General tab of a DOE-C package displays an invalid user account error for the Error State property.

Explanation: SAP servers could not authenticate this user with the Username and Password configured for this package.

User names and passwords configured for the connection pool cannot be tested before they are used. Errors are only reported after the connection fails. Errors typically occur during an administrative operation (such as unsubscribing a subscription), or in response to an asynchronous message for a subscription from DOE. On a system with existing DOE-C subscriptions, the initial resynchronization at startup would implicitly test the technical user.

Solution: Check the username and password configured for this user in the Connection Pool configured for the package. If it is incorrect, edit the properties used.

Note: If you change the username or password property of a DOE-C connection, you must reopen the same dialog and click `Test Connection` after saving. Otherwise the error state of this DOE-C package cannot be cleaned up. If you do not click `Test Connection`, the username or password is correct, but the error state of the DOE-C package cannot be cleaned up.

Application and Application User Management Issues

Review this list of documented issues for applications or application users managed by Sybase Control Center.

Wrong Application for Code Error

Problem: Application registration using a Windows Mobile emulator appears successful in Sybase Control Center, but the application log shows a `Wrong Application for Code` error when the application attempts to connect to Unwired Server.

This error occurs when you:

- Hard reset a Windows Mobile device emulator,
- Close an emulator without saving the emulator state, or
- Uninstall and reinstall the Unwired Server client software on the device.

Explanation: Because emulators do not generate unique application IDs, the Unwired Server messaging software on the device creates an application ID during installation and stores it in the emulator application registry. After registration, this permanent link between the emulator and the application ID must remain.

Hard resetting the emulator, closing the emulator without saving the emulator state, or uninstalling and reinstalling the Unwired Server client software purges the device registry and breaks the link between Unwired Server and the device software. When you attempt to reconnect, Unwired Server creates a new application ID for the device. Without the original application ID, the server cannot identify the device emulator, and therefore, cannot establish a relationship between the application and the activation code.

To avoid this problem so that the emulator and server remain synchronized, always save the emulator state before you close the emulator, and refrain from hard resetting the emulator, or uninstalling and reinstalling the client software.

Note: Before saving the state of an emulator, always uncradle the emulator using the Device Emulation Manager. This allows the device emulator to be cradled when the save image is loaded and used in the future.

Solution: Reconnect the emulator by either:

1. Deleting the original application from Unwired Server, then reregister the application, or
2. Reregistering the application

User Name of Registered Application Connection Not Displayed

Problem: The configured user name of a registered application connection is not displayed when you later review the properties for a device in Sybase Control Center. The **Application Connections** tab shows other properties but not the user name.

Explanation: The user name used for a application connection registration is not stored or handled as an application property.

Solution: To view the user name of the registered application in Sybase Control Center:

1. In the left navigation pane, click the **Applications** node.
2. In the right administration pane, click the **Application Users** tab.
3. In the table of registered users, for the user.
4. You can also select the **Application Connections** tab, and check the users properties.

Internal Server Error When Clicking Applications

Problem : Once logged into Sybase Control Center, the administrator clicks Applications in the navigation pane, and an Internal server error message is displayed.

After receiving this error, the administrator is further unable to register any applications because the **OK** button remains disabled.

Solution:

1. Validate the error:
 - a. Open `<UnwiredPlatform_InstallDir>\SCC-X_X\log\gateway.log`.
 - b. Look for this error: Caused by:
`com.sybase.uep.sysadmin.management.exception.ImoWsException: An error occurred loading a configuration file: Attempted to read or write protected memory. This is often an indication that other memory is corrupt.`
2. Validate that the Sybase Unwired Server service is running and there are no errors being reported in the Windows Application event log by that service.
3. Validate that the Messaging Server Administration Web Service is running:
 - a. Open a Web browser.
 - b. Open `http://localhost:5100/MobileOffice/admin.asmx`.
 - c. Select the **GetDeviceList2** method, then click **Invoke**.
 - d. Check whether a valid XML response returns.
4. If anything in steps 1-3 is unexpected, you may have an installation or configuration issue. Confirm this by:
 - a. Restarting the Sybase Unwired Server service.
 - b. Once available, repeat steps 2-3.

Troubleshoot the Sybase Control Center

- Otherwise, open Sybase Control Center, and click Applications to try registering an application again.
5. If you still get the same error and same behavior, contact Sybase Support.

Glossary

Defines terms used in Sybase Control Center documentation.

Glossary: Sybase Unwired Platform

Defines terms for all Sybase Unwired Platform components.

administration perspective – Or administration console. The Unwired Platform administrative perspective is the Flash-based Web application for managing Unwired Server. *See* Sybase Control Center.

administrators – Unwired Platform users to which an administration role has been assigned. A user with the "SUP Administrator" role is called a "platform administrator" and a user with the "SUP Domain Administrator" role is called a "domain administrator". These administration roles must also be assigned SCC administration roles to avoid having to authenticate to Sybase Control Center in addition to Unwired Server:

- A domain administrator only requires the "sccUserRole" role.
- A platform administrator requires both the "sccAdminRole" and "sccUserRole" roles.

Adobe Flash Player – Adobe Flash Player is required to run Sybase Control Center. Because of this player, you are required to run Sybase Control Center in a 32-bit browser. Adobe does not support 64-bit browsers.

Advantage Database Server[®] – A relational database management system that provides the messaging database for Sybase Unwired Platform. *See* messaging database.

Afaria[®] – An enterprise-grade, highly scalable device management solution with advanced capabilities to ensure that mobile data and devices are up-to-date, reliable, and secure. Afaria is a separately licensed product that can extend the Unwired Platform in a mobile enterprise. Afaria includes a server (Afaria Server), a database (Afaria Database), an administration tool (Afaria Administrator), and other runtime components, depending on the license you purchase.

application – In Unwired Server (and visible in Sybase Control Center), and application is the runtime entity that can be directly correlated to a native or mobile workflow application. The application definition on the server establishes the relationship among packages used in the application, domain that the application is deployed to, user activation method for the application, and other application specific settings.

APNS – Apple Push Notification Service.

application connection – A unique connection to the application on a device.

application connection template – a template for application connections that includes application settings, security configuration, domain details, and so forth.

application node – In Sybase Control Center, this is a registered application with a unique ID. This is the main entity that defines the behavior of device and backend interactions.

application registration – The process of registering an application with Sybase Unwired Platform. Registration requires a unique identity that defines the properties for the device and backend interaction with Unwired Server.

artifacts – Artifacts can be client-side or automatically generated files; for example: .xml, .cs, .java, .cab files.

availability – Indicates that a resource is accessible and responsive.

BAPI – Business Application Programming Interface. A BAPI is a set of interfaces to object-oriented programming methods that enable a programmer to integrate third-party software into the proprietary R/3 product from SAP®. For specific business tasks such as uploading transactional data, BAPIs are implemented and stored in the R/3 system as remote function call (RFC) modules.

BLOB – Binary Large Object. A BLOB is a collection of binary data stored as a single entity in a database management system. A BLOB may be text, images, audio, or video.

cache – The virtual tables in the Unwired Server cache database that store synchronization data. *See* cache database.

cache group – Defined in Unwired WorkSpace, MBOs are grouped and the same cache refresh policy is applied to their virtual tables (cache) in the cache database

cache partitions – Partitioning the cache divides it into segments that can be refreshed individually, which gives better system performance than refreshing the entire cache. Define cache partitions in Unwired WorkSpace by defining a partition key, which is a load argument used by the operation to load data into the cache from the enterprise information system (EIS).

cache database – Cache database. The Unwired Server cache database stores runtime metadata (for Unwired Platform components) and cache data (for MBOs). *See also* data tier.

CLI – Command line interface. CLI is the standard term for a command line tool or utility.

client application – *See* mobile application.

client object API – The client object API is described in the *Developer Guide: BlackBerry Native Applications*, *Developer Guide: iOS Native Applications*, and *Developer Guide: Windows and Windows Mobile Native Applications*.

cluster – Also known as a server farm. Typically clusters are setup as either runtime server clusters or database clusters (also known as a data tier). Clustering is a method of setting up redundant Unwired Platform components on your network in order to design a highly scalable and available system architecture.

cluster database – A data tier component that holds information pertaining to all Unwired Platform server nodes. Other databases in the Unwired Platform data tier includes the cache, messaging, and monitoring databases.

connection – Includes the configuration details and credentials required to connect to a database, Web service, or other EIS.

connection pool – A connection pool is a cache of Enterprise Information System (EIS) connections maintained by Unwired Server, so that the connections can be reused when Unwired Server receives future requests for data.

For proxy connections, a connection pool is a collection of proxy connections pooled for their respective back-ends, such as SAP Gateway.

connection profile – In Unwired WorkSpace, a connection profile includes the configuration details and credentials required to connect to an EIS.

context variable – In Unwired WorkSpace, these variables are automatically created when a developer adds reference(s) to an MBO in a mobile application. One table context variable is created for each MBO attribute. These variables allow mobile application developers to specify form fields or operation parameters to use the dynamic value of a selected record of an MBO during runtime.

data change notification (DCN) – Data change notification (DCN) allows an Enterprise Information System (EIS) to synchronize its data with the cache database through a push event.

data refresh – A data refresh synchronizes data between the cache database and a back-end EIS so that data in the cache is updated. *See also* scheduled data refresh.

data source – In Unwired WorkSpace, a data source is the persistent-storage location for the data that a mobile business object can access.

data tier – The data tier includes Unwired Server data such as cache, cluster information, and monitoring. The data tier includes the cache database (CDB), cluster, monitoring, and messaging databases.

data vault – A secure store across the platform that is provided by an SUP client.

deploy – (Unwired Server) Uploading a deployment archive or deployment unit to an Unwired Server instance. Unwired Server can then make these units accessible to users via a client application that is installed on a mobile device.

There is a one-to-one mapping between an Unwired WorkSpace project and a server package. Therefore, all MBOs that you deploy from one project to the same server are deployed to the same server package.

deployment archive – In Unwired WorkSpace, a deployment archive is created when a developer creates a package profile and executes the **build** operation. Building creates an archive that contains both a deployment unit and a corresponding descriptor file. A

deployment archive can be delivered to an administrator for deployment to a production version of Unwired Server.

deployment descriptor – A deployment descriptor is an XML file that describes how a deployment unit should be deployed to Unwired Server. A deployment descriptor contains role-mapping and domain-connection information. You can deliver a deployment descriptor and a deployment unit—jointly called a deployment archive—to an administrator for deployment to a production version of Unwired Server.

deployment mode – You can set the mode in which a mobile application project or mobile deployment package is deployed to the target Unwired Server.

deployment profile – A deployment profile is a named instance of predefined server connections and role mappings that allows developers to automate deployment of multiple packages from Sybase Unwired WorkSpace to Unwired Server. Role mappings and connection mappings are transferred from the deployment profile to the deployment unit and the deployment descriptor.

deployment unit – The Unwired WorkSpace build process generates a deployment unit. It enables a mobile application to be effectively installed and used in either a preproduction or production environment. Once generated, a deployment unit allows anyone to deploy all required objects, logical roles, personalization keys, and server connection information together, without requiring access to the whole development project. You can deliver a deployment unit and a deployment descriptor—jointly called a deployment archive—to an administrator for deployment to a production version of Unwired Server.

development package – A collection of MBOs that you create in Unwired WorkSpace. You can deploy the contents of a development package on an instance of Unwired Server.

device application – *See also* mobile application. A device application is a software application that runs on a mobile device.

device notification – Replication synchronization clients receive device notifications when a data change is detected for any of the MBOs in the synchronization group to which they are subscribed. Both the change detection interval of the synchronization group and the notification threshold of the subscription determine how often replication clients receive device notifications. Administrators can use subscription templates to specify the notification threshold for a particular synchronization group.

device user – The user identity tied to a device.

DML – Data manipulation language. DML is a group of computer languages used to retrieve, insert, delete, and update data in a database.

DMZ – Demilitarized zone; also known as a perimeter network. The DMZ adds a layer of security to the local area network (LAN), where computers run behind a firewall. Hosts running in the DMZ cannot send requests directly to hosts running in the LAN.

domain administrator – A user to which the platform administrator assigns domain administration privileges for one or more domain partitions. The domain administrator has a restricted view in Sybase Control Center, and only features and domains they can manage are visible.

domains – Domains provide a logical partitioning of a hosting organization's environment, so that the organization achieves increased flexibility and granularity of control in multitenant environments. By default, the Unwired Platform installer creates a single domain named "default". However the platform administrator can also add more domains as required.

EIS – Enterprise Information System. EIS is a back-end system, such as a database.

Enterprise Explorer – In Unwired WorkSpace, Enterprise Explorer allows you to define data source and view their metadata (schema objects in case of database, BAPIs for SAP, and so on).

export – The Unwired Platform administrator can export the mobile objects, then import them to another server on the network. That server should meet the requirement needed by the exported MBO.

hostability – *See* multitenantcy.

IDE – Integrated Development Environment.

JDE – BlackBerry Java Development Environment.

key performance indicator (KPI) – Used by Unwired Platform monitoring. KPIs are monitoring metrics that are made up for an object, using counters, activities, and time which jointly for the parameters that show the health of the system. KPIs can use current data or historical data.

keystore – The location in which encryption keys, digital certificates, and other credentials in either encrypted or unencrypted keystore file types are stored for Unwired Server runtime components. *See also* truststore.

LDAP – Lightweight Directory Access Protocol.

local business object – Defined in Unwired WorkSpace, local business objects are not bound to EIS data sources, so cannot be synchronized. Instead, they are objects that are used as local data store on device.

logical role – Logical roles are defined in mobile business objects, and mapped to physical roles when the deployment unit that contain the mobile business objects are deployed to Unwired Server.

matching rules – A rule that triggers a mobile workflow application. Matching rules are used by the mobile workflow email listener to identify e-mails that match the rules specified by the administrator. When emails match the rule, Unwired Server sends the e-mail as a mobile workflow to the device that matches the rule. A matching rule is configured by the administrator in Sybase Control Center.

MBO – Mobile business object. The fundamental unit of data exchange in Sybase Unwired Platform. An MBO roughly corresponds to a data set from a back-end data source. The data can come from a database query, a Web service operation, or SAP. An MBO contains both concrete implementation-level details and abstract interface-level details. At the implementation-level, an MBO contains read-only result fields that contain metadata about the data in the implementation, and parameters that are passed to the back-end data source. At the interface-level, an MBO contains attributes that map to result fields, which correspond to client properties. An MBO may have operations, which can also contain parameters that map to arguments, and which determines how the client passes information to the enterprise information system (EIS).

You can define relationships between MBOs, and link attributes and parameters in one MBO to attributes and parameters in another MBO.

MBO attribute – An MBO attribute is a field that can hold data. You can map an MBO attribute to a result field in a back-end data source; for example, a result field in a database table.

MBO binding – An MBO binding links MBO attributes and operations to a physical data source through a connection profile.

MBO operation – An MBO operation can be invoked from a client application to perform a task; for example, create, delete, or update data in the EIS.

MBO relationship – MBO relationships are analogous to links created by foreign keys in a relational database. For example, the account MBO has a field called *owner_ID* that maps to the *ID* field in the owner MBO.

Define MBO relationships to facilitate:

- Data synchronization
- EIS data-refresh policy

messaging based synchronization – A synchronization method where data is delivered asynchronously using a secure, reliable messaging protocol. This method provides fine-grained synchronization (synchronization is provided at the data level—each process communicates only with the process it depends on), and it is therefore assumed that the device is always connected and available. *See also* synchronization.

messaging database – The messaging database allows in-flight messages to be stored until they can be delivered. This database is used in a messaging based synchronization environment. The messaging database is part of the Unwired Platform data tier, along with the cache, cluster, and monitoring databases.

mobile application – A Sybase Unwired Platform mobile application is an end-to-end application, which includes the MBO definition (back-end data connection, attributes, operations, and relationships), the generated server-side code, and the client-side application code.

Mobile Application Diagram – The Mobile Application Diagram is the graphical interface to create and edit MBOs. By dragging and dropping a data source onto the Mobile Application Diagram, you can create a mobile business object and generate its attribute mappings automatically.

Mobile Application Project – A collection of MBOs and client-side, design-time artifacts that make up a mobile application.

mobile workflow packages – Mobile workflow packages use the messaging synchronization model. The mobile workflow packages are deployed to Unwired Server, and can be deployed to mobile devices, via the Unwired Platform administrative perspective in Sybase Control Center.

monitoring – Monitoring is an Unwired Platform feature available in Sybase Control Center that allows administrators to identify key areas of weakness or periods of high activity in the particular area they are monitoring. It can be used for system diagnostic or for troubleshooting. Monitored operations include replication synchronization, messaging synchronization, messaging queue, data change notification, device notification, package, user, and cache activity.

monitoring database – A database that exclusively stores data related to replication and messaging synchronization, queues status, users, data change notifications, and device notifications activities. By default, the monitoring database runs in the same data tier as the cache database, messaging database and cluster database.

monitoring profiles – Monitoring profiles specify a monitoring schedule for a particular group of packages. These profiles let administrators collect granular data on which to base domain maintenance and configuration decisions.

multitenancy – The ability to host multiple tenants in one Unwired Cluster. Also known as hostability. *See also* domains.

node – A host or server computer upon which one or more runtime components have been installed.

object query – Defined in Unwired WorkSpace for an MBO and used to filter data that is downloaded to the device.

onboarding – The enterprise-level activation of an authentic device, a user, and an application entity as a combination, in Unwired Server.

operation – *See* MBO operation.

package – A package is a named container for one or more MBOs. On Unwired Server a package contains MBOs that have been deployed to this instance of the server.

palette – In Unwired WorkSpace, the palette is the graphical interface view from which you can add MBOs, local business objects, structures, relationships, attributes, and operations to the Mobile Application Diagram.

parameter – A parameter is a value that is passed to an operation/method. The operation uses the value to determine the output. When you create an MBO, you can map MBO parameters to data-source arguments. For example, if a data source looks up population based on a state abbreviation, the MBO gets the state from the user, then passes it (as a parameter/argument) to the data source to retrieve the information. Parameters can be:

- Synchronization parameters – synchronize a device application based on the value of the parameter.
- Load arguments – perform a data refresh based on the value of the argument.
- Operation parameters – MBO operations contain parameters that map to data source arguments. Operation parameters determine how the client passes information to the enterprise information system (EIS).

personalization key – A personalization key allows a mobile device user to specify attribute values that are used as parameters for selecting data from a data source. Personalization keys are also used as operation parameters. Personalization keys are set at the package level. There are three type of personalization keys: Transient, client, server.

They are most useful when they are used in multiple places within a mobile application, or in multiple mobile applications on the same server. Personalization keys may include attributes such as name, address, zip code, currency, location, customer list, and so forth.

perspective – A named tab in Sybase Control Center that contains a collection of managed resources (such as servers) and a set of views associated with those resources. The views in a perspective are chosen by users of the perspective. You can create as many perspectives as you need and customize them to monitor and manage your resources.

Perspectives allow you to group resources ways that make sense in your environment—by location, department, or project, for example.

physical role – A security provider group or role that is used to control access to Unwired Server resources.

Problems view – In Eclipse, the Problems view displays errors or warnings for the Mobile Application Project.

provisioning – The process of setting up a mobile device with required runtimes and device applications. Depending on the synchronization model used and depending on whether or not the device is also an Afaria client, the files and data required to provision the device varies.

pull synchronization – Pull synchronization is initiated by a remote client to synchronize the local database with the cache database. On Windows Mobile, pull synchronization is supported only in replication applications.

push synchronization – Push is the server-initiated process of downloading data from Unwired Server to a remote client, at defined intervals, or based upon the occurrence of an event.

queue – In-flight messages for a messaging application are saved in a queue. A queue is a list of pending activities. The server then sends messages to specific destinations in the order that

they appear in the queue. The depth of the queue indicates how many messages are waiting to be delivered.

relationship – *See* MBO relationship.

relay server – *See also* Sybase Hosted Relay Service.

resource – A unique Sybase product component (such as a server) or a subcomponent.

REST web services – Representational State Transfer (REST) is a style of software architecture for distributed hypermedia systems such as the World Wide Web.

RFC – Remote Function Call. You can use the RFC interface to write applications that communicate with SAP R/3 applications and databases. An RFC is a standalone function. Developers use SAP tools to write the Advanced Business Application Programming (ABAP) code that implements the logic of a function, and then mark it as "remotely callable," which turns an ABAP function into an RFC.

role – Roles control access to Sybase Unwired Platform resources. *See also* logical role and physical role.

role mapping – Maps a physical (server role) to a logical (Unwired Platform role). Role mappings can be defined by developers, when they deploy an MBO package to a development Unwired Server, or by platform or domain administrators when they assign a security configuration to a domain or deploy a package to a production Unwired Server (and thereby override the domain-wide settings in the security configuration).

RSOE – Relay Server Outbound Enabler. An RSOE is an application that manages communication between Unwired Server and a relay server.

runtime server – An instance of Unwired Server that is running. Typically, a reference to the runtime server implies a connection to it.

SAP – SAP is one of the EIS types that Unwired Platform supports.

SCC – Sybase Control Center. A Web-based interface that allows you to administer your installed Sybase products.

schedule – The definition of a task (such as the collection of a set of statistics) and the time interval at which the task must execute in Sybase Control Center.

scheduled data refresh – Data is updated in the cache database from a back-end EIS, based on a scheduled data refresh. Typically, data is retrieved from an EIS (for example, SAP) when a device user synchronizes. However, if an administrator wants the data to be preloaded for a mobile business object, a data refresh can be scheduled so that data is saved locally in a cache. By preloading data with a scheduled refresh, the data is available in the information server when a user synchronizes data from a device. Scheduled data refresh requires that an administrator define a cache group as "scheduled" (as opposed to "on-demand").

security configuration – Part of the application user and administration user security. A security configuration determines the scope of user identity, authentication and authorization

checks, and can be assigned to one or more domains by the platform administrator in Sybase Control Center. A security configuration contains:

- A set of configured security providers (for example LDAP) to which authentication, authorization, attribution is delegated.
- Role mappings (which can be specified at the domain or package level)

security provider – A security provider and its repository holds information about the users, security roles, security policies, and credentials used by some to provide security services to Unwired Platform. A security provider is part of a security configuration.

security profile – Part of the Unwired Server runtime component security. A security profile includes encryption metadata to capture certificate alias and the type of authentication used by server components. By using a security profile, the administrator creates a secured port over which components communicate.

server connection – The connection between Unwired WorkSpace and a back-end EIS is called a server connection.

server farm – *See also* cluster. Is the relay server designation for a cluster.

server-initiated synchronization – *See* push synchronization.

SOAP – Simple Object Access Protocol. SOAP is an XML-based protocol that enables applications to exchange information over HTTP. SOAP is used when Unwired Server communicates with a Web service.

solution – In Visual Studio, a solution is the high-level local workspace that contains the projects users create.

Solution Explorer – In Visual Studio, the Solution Explorer pane displays the active projects in a tree view.

SSO – Single sign-on. SSO is a credential-based authentication mechanism.

statistics – In Unwired Platform, the information collected by the monitoring database to determine if your system is running as efficiently as possible. Statistics can be current or historical. Current or historical data can be used to determine system availability or performance. Performance statistics are known as key performance indicators (KPI).

Start Page – In Visual Studio, the Start Page is the first page that displays when you launch the application.

structured data – Structured data can be displayed in a table with columns and labels.

structure object – Defined in Unwired WorkSpace, structures hold complex datatypes, for example, a table input to a SAP operation.

subscription – A subscription defines how data is transferred between a user's mobile device and Unwired Server. Subscriptions are used to notify a device user of data changes, then these updates are pushed to the user's mobile device.

Sybase Control Center – Sybase Control Center is the Flash-based Web application that includes a management framework for multiple Sybase server products, including Unwired Platform. Using the Unwired Platform administration perspective in Sybase Control Center, you can register clusters to manage Unwired Server, manage domains, security configurations, users, devices, connections, as well as monitor the environment. You can also deploy and MBO or workflow packages, as well as register applications and define templates for them. Only use the features and documentation for Unwired Platform. Default features and documentation in Sybase Control Center do not always apply to the Unwired Platform use case.

Sybase Control Center X.X Service – Provides runtime services to manage, monitor, and control distributed Sybase resources. The service must be running for Sybase Control Center to run. Previously called Sybase Unified Agent.

Sybase Hosted Relay Service – The Sybase Hosted Relay Service is a Web-hosted relay server that enables you to test your Unwired Platform development system.

Sybase Messaging Service – The synchronization service that facilitates communication with device client applications.

Sybase Unwired Platform – Sybase Unwired Platform is a development and administrative platform that enables you to mobilize your enterprise. With Unwired Platform, you can develop mobile business objects in the Unwired WorkSpace development environment, connect to structured and unstructured data sources, develop mobile applications, deploy mobile business objects and applications to Unwired Server, which manages messaging and data services between your data sources and your mobile devices.

Sybase Unwired WorkSpace – Sybase Unwired Platform includes Unwired WorkSpace, which is a development tool for creating mobile business objects and mobile applications.

synchronization – A synchronization method where data is delivered synchronously using an upload/download pattern. For push-enabled clients, synchronization uses a "poke-pull" model, where a notification is pushed to the device (poke), and the device fetches the content (pull), and is assumed that the device is not always connected to the network and can operate in a disconnected mode and still be productive. For clients that are not push-enabled, the default synchronization model is pull. *See also* messaging based synchronization.

synchronization group – Defined in Unwired WorkSpace, a synchronization group is a collection of MBOs that are synchronized at the same time.

synchronization parameter – A synchronization parameter is an MBO attribute used to filter and synchronize data between a mobile device and Unwired Server.

synchronization phase – For replication based synchronization packages, the phase can be an upload event (from device to the Unwired Server cache database) or download event (from the cache database to the device).

Glossary

synchronize – *See also* data refresh. Synchronization is the process by which data consistency and population is achieved between remote disconnected clients and Unwired Server.

truststore – The location in which certificate authority (CA) signing certificates are stored. *See also* keystore.

undeploy – Running **undeploy** removes a domain package from an Unwired Server.

Unwired Server – The application server included with the Sybase Unwired Platform product that manages mobile applications, back-end EIS synchronization, communication, security, transactions, and scheduling.

user – Sybase Control Center displays the mobile-device users who are registered with the server.

view – A window in a perspective that displays information about one or more managed resources. Some views also let you interact with managed resources or with Sybase Control Center itself. For example, the Perspective Resources view lists all the resources managed by the current perspective. Other views allow you to configure alerts, view the topology of a replication environment, and graph performance statistics.

Visual Studio – Microsoft Visual Studio is an integrated development environment product that you can use to develop device applications from generated Unwired WorkSpace code.

Welcome page – In Eclipse, the first set of pages that display when you launch the application.

workspace – In Eclipse, a workspace is the directory on your local machine where Eclipse stores the projects that you create.

WorkSpace Navigator – In Eclipse, the tree view that displays your mobile application projects.

WSDL file – Web Service Definition Language file. The file that describes the Web service interface that allows clients to communicate with the Web service. When you create a Web service connection for a mobile business object, you enter the location of a WSDL file in the URL.

Index

- Xmx maximum memory option 15
- XX:MaxPermSize permanent memory option 15

A

- accessibility 20
- Active Directory
 - use considerations 152
- administration
 - core administration nodes 2
 - administration listener 45
 - administration performance 252
 - administration perspective
 - empty SCC console screen 253
 - gray SCC console screen 253
 - rectangular box instead of SCC console 253
 - administration users
 - configuring 78, 133
 - maintaining 78, 133
 - administrators
 - login accepted when authentication provider unavailable 258
- Adobe Flex 20
- agent.log file 254
- Alert Message property 179
- alerts
 - effects of repository purging on history 26
- Alerts property 179
- alias, certificate 131
- Allow Roaming property 181
- Apache
 - relay server configuration 39
- Apache logs 67
- APNS Device Token property 179
- appenders, adding 67
- Apple push notification properties 179
- Apple push notification, configuring 54
- application 168
- application connection
 - activation options 169
- application connection template 177, 178
- application connection templates
 - administration overview 5
- application connections
 - administration overview 5

- application creation 167
- application creation wizard 167
- application ID
 - guidelines 173
 - overview 168
- application management issues 272
- application settings 179
- application user management issues 272
- application users 170
- applications 166, 167, 171
 - administration overview 5
- Application Connection properties 176
- authentication
 - DCNs 201
 - provider unavailable but administrator can log in 258
- authentication cache timeout 137
- authentication failure 102
- AuthenticationScope 153
- AuthorizationException 269
- auto purge
 - monitoring data 224
 - removing domain log data 262
 - removing monitor data 262
- automated message processing 181
- automatic registration 179

B

- backups
 - about 22
 - changing the schedule 24
 - forcing 24
 - restoring from 25
 - scheduling 23
 - suspending and resuming 24
- Badges property 179
- BlackBerry push notification, configuring 55

C

- cache 192
- cache group
 - configuring 192
 - purging 197

Index

- status statistics 248
 - cache interval
 - real time 197
 - cache monitoring 246
 - cache performance statistics 247
 - cache refresh
 - custom 195
 - daily 194
 - hourly 194
 - never schedule 197
 - on demand 193
 - scheduling 194
 - cache statistics
 - viewing 246
 - cache timeout, setting 137
 - caching of login credentials 269
 - cannot start Unwired Server 265
 - CDB 192
 - certificate alias 131
 - certificate problems 257
 - CertificateAuthenticationLoginModule
 - authentication module
 - for SAP single sign-on and X.509 163, 165
 - certificates
 - for context variables 214
 - managing for RSOE 60
 - cleaning up the Flash Player cache 11, 259
 - client application logs
 - checking 108
 - cleaning 108
 - cluster properties 32
 - clusters 31
 - administration overview 3
 - affected by configuration changes 31
 - status, checking 34
 - communication ports
 - communication port properties, configuring 46
 - security configuration 46
 - SSL encryption 46, 47
 - configuration changes
 - effect on clusters 31
 - configuration files
 - Relay Server 35
 - connection errors 267
 - connection templates, creating 109
 - Connections 85
 - connections between Unwired Server and data sources 173
 - connections, creating 109
 - console
 - about 27
 - commands 27
 - context variables 216
 - configuring 214
 - correlating log data across subsystems 95
 - creating applications 167
 - credentials
 - old, ability to authenticate with 269
 - custom settings for messaging devices 181
- ## D
- data cache
 - cache 192
 - data change notification monitoring
 - histories 239
 - performance statistics 240
 - data change notification statistics 239
 - data sources
 - connections to 173
 - databases
 - monitoring 223
 - DCN log data
 - general DCN 101
 - workflow DCN 101
 - DCNRole 201
 - Debug Trace Level property 181
 - Debug Trace Size property 181
 - degrading performance 270
 - deleting 178
 - Delivery Threshold property 179
 - deploy failure for large packages 270
 - deployment
 - mobile workflow archives 185
 - package archives 185
 - deployment issues for packages 270
 - device log error 272
 - Device Log Items property 181
 - device notification
 - history statistics 241
 - performance statistics 241
 - device notification log data 101
 - device notification monitoring 240, 241
 - device notifications
 - configuring 203
 - statistics 240
 - Device Subtype property 182
 - device user name not displayed 273

device users
 assigning mobile workflow packages 216

devices
 Apple push notification properties 179
 user assignments 173

disabling security with NoSec 144

DNS server failure 255

documentation roadmap 1

DOE-C
 invalid user 210, 271

DOE-C packages 209

domain 78
 security configuration, choosing 133
 security, assigning security configuration 166
 security, configuring 132

domain administrators
 registering 77

domain log data
 not displayed 262

domain logs 83

domains 75
 creating 76
 deleting 77
 enabling 76

E

e-mail
 redirecting with matching rules 213

editing an application connection 176

EIS
 connection properties 111

Enable property 179

enabling log profile 86

enterprise information systems
 See EIS

environment variables
 SCC_MEM_MAX 14, 15
 SCC_MEM_PERM 14, 15

error messages
 logging levels 68
 server logs 67

errors
 user account failure 210, 271

errors log data 103

export failure for large packages 270

exporting log data 96

F

F5 (browser refresh)
 logging out of Sybase Control Center 257

Flash Player 12
 cleaning up the cache after upgrade 11, 259

flush batch size for monitoring data 224

flush threshold for monitoring data 224

full backups 23

G

gateway.log file 265

general application properties 168

getting started after installing 12

glossaries 275
 Sybase Unwired Platform terms 275

H

hard coded credentials 216

heat chart
 launch icon 18

help command (console) 27

help system, accessing 12

History tab is blank 261

host name changes not reflected in SCC 259

host name resolution failure 255

HTTPS
 RSOE certificates 60

I

icons
 in SCC toolbar 18

IIS
 relay server configuration 39

IMSI property 182

incremental backups 23

info command (console) 27

invalid login 256

iOS push notification properties 179

J

Java system properties
 displaying information about 27

JDBC properties 111

jvmpot memory options for Windows services 14,
 15

Index

K

- Keep Alive (sec) property 181
- keyboard shortcuts for Adobe Flex 20

L

- LDAP
 - configuration properties 144
 - processes 256
 - role computation 153
 - stacking providers 141, 154
 - startup 256
- LDAP security provider
 - modules available 152
- LDAP trees
 - multiple 153
- licenses
 - servers, reviewing 33
- listener startup errors 267
- log files
 - agent.log file 254
 - scc_repository.log 254
 - server logs 68
- log filters 89
- log profile 86
- log, server
 - refreshing 71
- log4j.xml 67
- logging in to Sybase Control Center
 - clearing authentication parameters 258
- logging in to Sybase Control Center - first user 12
- logging levels 68
- logging out of a server 13
- logging out of Sybase Control Center
 - unintentionally, using F5 browser refresh 257
- logical roles
 - DCNs 201
- login accounts, default
 - about 12
- login invalid 256
- login session timeout
 - setting 15
- login troubleshooting
 - Sybase Control Center 258
- logs
 - client application 108
 - DOE connections 94
 - domain-level 83
 - general DCN 92

- JDBC connections 94
- life cycles 68
- outbound enabler 63
- REST connections 94
- SAP connections 94
- server 67
- server, configuring 68
- SOAP connections 94
- synchronization 91
- synchronization logs 91
- Unwired Server 67
- workflow DCN 92

M

- management console unavailable 259
- management issues for packages 270
- Managing properties 178
- manual control of message processing 181
- Manual registration 167
- matching rules for redirecting e-mail
 - configuring 213
 - testing 214
- MBO create error history 209
- MBO data
 - See cache
- MBO delete error history 209
- MBO error history 208
- MBO operation error history 209
- MBO status statistics 247
- MBO update error history 209
- memory
 - configuring 14
 - displaying information about 27
- messaging 53
 - configuring properties 54
 - configuring subscriptions 205
- messaging device advanced properties 181
- messaging device connection properties 180
- messaging devices
 - custom settings 181
 - information properties 182
- messaging history monitoring
 - detail view 235
 - summary view 235
- messaging monitoring
 - history 235
 - performance statistics 237
 - request statistics 234

- messaging packages
 - statistics 243
 - messaging queue counts 270
 - messaging queues
 - statistics 238
 - status data 238
 - messaging statistics 234
 - messaging synchronization
 - monitoring 235
 - messaging users
 - monitoring 245
 - Microsoft Cluster issues 259
 - mobile business objects
 - cache group status statistics 248
 - clearing error history 208
 - connections to 173
 - reviewing error history 208
 - mobile devices
 - properties identifying 182
 - mobile workflow packages
 - assigning device users 216
 - configuring 212
 - configuring notification mailbox 211
 - deploying 218
 - mobile workflows 211
 - checking users and queues 217
 - configuring display name and icon 212
 - mobile workflow packages administration 10
 - Model property 182
 - monitoring 227
 - cache 246
 - cache group status 248
 - cache performance 247
 - data change notification statistics 239
 - database, configuring 224
 - device notification history 241
 - device notification performance 241
 - device notifications 240
 - issues for platform components 260
 - MBO status 247
 - messaging queue statistics 238
 - messaging statistics 234
 - messaging synchronization 235
 - messaging user statistics 245
 - replication statistics 230
 - replication user statistics 244
 - replication-based synchronization 231
 - statistic categories 229
 - user security 230
 - user statistics 244
 - monitoring data 227
 - auto purge 224
 - exporting 228
 - flush batch size 224
 - flush threshold 224
 - not displayed 262
 - purging 228
 - reviewing 227
 - searching 229
 - monitoring profiles 225
 - creating and enabling 225
 - monitoring schedule
 - custom 226
 - monitoring setup
 - effect on clusters 31
 - monitoring Unwired Platform 221
 - overview 8, 222
 - multiple LDAP trees 153
- ## N
- named security configuration
 - domain, selecting 133
 - Navigating applications 170
 - notification mailbox 211
- ## O
- onboarding 5
 - operation error history 209
 - Outbound Enabler 59
 - configuration 63
 - configuring 60, 61, 63, 64
 - custom configuration 63
 - deleting configuration 64
 - downloading log files 63
 - generate configuration file 38, 62
 - information in SCC 66
 - list of 65
 - loading certificates 60
 - logging options 62
 - managing 63
 - setting up 60
 - starting and stopping 44, 65
 - startup options 61, 62, 64

Index

P

- package deployment and management issues 270
- package statistics 242
- package subscriptions
 - configuring 204
 - managing 206
 - pinging 206
 - recovering 206
 - resuming 206
 - resynchronizing 206
 - suspending 206
 - unsubscribing 206
- packages
 - administration overview 6
 - cache properties 193
 - contents, exporting 190
 - contents, importing 190
 - enabling and disabling 191, 209
 - mobile workflow administration overview 10
 - replication based synchronization 190
 - security 198
- PAP, Push Access Protocol 55
- passwords
 - old, ability to authenticate with 269
- payloads 86
- performance
 - setting BES MDS HTTP traffic 55
 - Sybase Control Center 11, 259
- performance data
 - administration 252
- performance degradation 270
- Perspective Resources view
 - about 20
 - show/hide icon 18
- perspectives
 - about 20
 - creating 21
 - removing 21
 - renaming 21
- Phone Number property 182
- pinging a server 44
- platform component monitoring issues 260
- port conflicts
 - among multiple SCC versions 255
 - with Sybase Control Center X.X 267
- port numbers 256
- ports
 - displaying information about 27
- postinstallation tasks 12

- problems starting Sybase Control Center services 254
- problems with application and application user management 272
- production edition 33
- profile definitions 83
- properties
 - advanced, of messaging devices 181
 - connection reference 111
 - custom settings for messaging devices 181
 - information on messaging devices 182
 - mobile workflows 212
 - monitoring database 223
 - package subscriptions, configuring 204
 - push notification for iOS 179
 - security provider configuration 142
- proxy 94
- Proxy 95
- proxy properties 132
- proxy push log data 107
- proxy request-response log data 107
- purging a cache group 197
- purging domain logs 96
- Push 95
- Push Access Gateway 55
- push notification properties for iOS 179

Q

- queue counts 270
- queues
 - messaging, status data 238
 - mobile workflow, checking 217

R

- Refreshing application view 171
- registered application 78
- reinstalling Sybase Control Center service 255
- Relay Server
 - backend farm configuration 37, 40
 - configuration properties in Relay Server tab 42
 - configuring 35
 - custom configuration 36, 39
 - deleting configuration 41
 - generate configuration 39
 - generate RSOE configuration 38, 62
 - information in SCC 42

- managing 39
 - outbound enabler 59
 - properties, viewing or editing 39
 - quick configuration 35
 - Relay Server Outbound Enabler 59, 60
 - configuration 63
 - configuring 60, 61, 63, 64
 - custom configuration 63
 - deleting configuration 64
 - downloading log files 63
 - generate configuration file 38, 62
 - information in SCC 66
 - list of 65
 - loading certificates 60
 - logging options 62
 - managing 63
 - setting up 60
 - starting and stopping 44, 65
 - startup options 61, 62, 64
 - Relay Server URL Prefix property 181
 - Relay Servers 34
 - replication history monitoring
 - detail view 231
 - summary view 231
 - replication monitoring
 - history 231
 - performance statistics 233
 - request statistics 230
 - replication packages
 - configuring subscriptions 203
 - statistics 242
 - replication statistics 230
 - replication subscription templates 203
 - replication synchronization 50
 - replication users
 - monitoring 244
 - replication-based synchronization
 - monitoring 231
 - repository 22
 - backing up 24
 - changing backup schedule 24
 - configuring purging 26
 - restoring from backup 25
 - scheduling backups 23
 - Request Response 94
 - resource explorer
 - launch icon 18
 - resources
 - remote, registering manually 17
 - restarting a remote server
 - unsuccessful 266
 - restarts, configuring in Windows 10
 - retrieving logs 88
 - roles
 - computing 153
 - overview 165
 - RSOE 60
 - rules for redirecting e-mail
 - configuring 213
 - testing 214
- S**
- SAP
 - user account error 210, 271
 - SAP connection properties 128
 - SAP DOE-C connections 128
 - SAP DOE-C properties 128
 - SAP single sign-on
 - SAPSSOTokenLoginModule authentication
 - module 161
 - SAPSSOTokenLoginModule authentication
 - properties 159
 - stacking login modules 140
 - SAP single sign-on with X.509
 - CertificateAuthenticationLoginModule
 - authentication module 163, 165
 - SAP/R3 properties 124
 - SAPSSOTokenLoginModule authentication
 - module
 - for SAP single sign-on 161
 - properties 159
 - SCC console tree is not complete 269
 - SCC_MEM_MAX 14, 15
 - SCC_MEM_PERM 14, 15
 - scc_repository.log file 254
 - searching 171, 173
 - Secure Sockets Layer encryption
 - communication ports 46
 - secure synchronization port 51
 - security
 - administration overview 8
 - domain, assigning security configuration 166
 - domain, configuring 132
 - monitoring 230
 - security certificates
 - See SSL certificates
 - security configuration
 - choosing 133

Index

- effect on clusters 31
- packages 198
- removing 133
- security configuration, creating 137
- security configurations
 - overview 136
- security error when connecting to SCC 258
- security log data 102
- security profile
 - communication port 46
 - management port 46
 - SSL certificates 47
- security profiles 47
 - communication port 47
 - management port 47
- security provider configuration properties 142
- security providers, reordering 142
- server
 - status 44
- server configuration
 - applying changes 58
 - effect on clusters 31
 - system performance properties 56
- server licensing 33
- server log
 - deleting 71
 - searching 70
- server ports
 - general 45
- server ports, viewing 59
- server properties 59
- server tier administration issues 263
- servers
 - log, refreshing 71
 - logging out of 13
 - logs, configuring 68
 - pinging 44
 - server properties 45
 - stopping and starting 43
 - suspending and resuming 44
- services
 - listing 27
- services, Windows
 - configuring SCC memory options for 15
 - running Sybase Control Center as 10
- setting BES MDS HTTP traffic 55
- shared data folder, asynchronous operation replays 32
- shutdown command (console) 28
- SOAP Web Services properties 131
- Sounds property 179
- SSL
 - mutual authentication 131
 - RSOE certificates 60
- SSL certificates 12, 47
 - error when missing 258
 - setting up 12
- SSL encryption
 - communication ports 46
 - security profile 47
- SSL keystore 47
- SSL truststore 47
- stacking LDAP modules 141, 154
- stacking login modules
 - for SAP single sign-on 140
- start up, automatic, configuring in Windows 10
- starting a remote server
 - unsuccessful 266
- starting servers 43
- statistics
 - application connection security 102
 - for messaging packages 243
 - for replication packages 242
- statistics chart
 - effects of repository purging on 26
- status command (console) 29
- stopping a remote server
 - unsuccessful 266
- stopping servers 43
- subscription templates
 - configuring for replication packages 203
 - creating 203
- subscriptions, DOE-C
 - reviewing 211
- SUP DCN User 201
- SupCertificateIssuer 216
- SupCertificateNotAfter 216
- SupCertificateNotBefore 216
- SupCertificateSubject 216
- SupPassword 216
 - for context variables 214
- SupUser 216
 - for context variables 214
- Sybase Control Center
 - about 1
 - accessibility 20
 - connecting a browser to 12
 - console commands 27

- dependence on Sybase Control Center X.X 10
- failure to start 255
- functionality not applicable to Unwired Platform 19
- logging out unintentionally with F5 257
- management tier issues 253
- reinstalling the service 255
- second version fails to start 255
- security error when connecting 258
- service port conflicts 267
- setting up SSL certificates 12
- starting in Windows 10
- starting in Windows as a service 10
- stopping in Windows 10
- Windows service fails to start 254
- Sybase Control Center performance 11, 259
- Sybase Control Center service 256
- sync group
 - configuring 192
- synchronization
 - configuring general properties 51
- synchronization listener properties 51
- synchronization log data
 - cache refresh 97
 - data services interface 97
 - data synchronization 97
 - operation replay 97
 - result checker 97
 - subscription 97
- synchronization port 51
- synchronization problems 265
- system data, reviewing 227
- system licensing 33
- system performance properties, configuring 56
- system properties
 - displaying information about 27

T

- TCP/IP filtering causing errors 267
- terms
 - Sybase Unwired Platform 275
- timeout
 - setting for login sessions 15
- toolbar icons 18
- troubleshooting
 - authentication failure 102
 - Outbound Enabler 63
 - Relay Server Outbound Enabler 63
 - Sybase Control Center problems 251

- Unwired Server problems 265
- Unwired Server startup 265
- user account failure 271
- troubleshooting Microsoft Cluster 259
- troubleshooting performance issues 252
- troubleshooting Unwired Platform with SCC 251

U

- uafshutdown.bat 10
- uafstartup.bat 10
- Unwired Platform
 - configuring 31
 - monitoring 8, 222
- Unwired Platform console
 - opening 16
- Unwired Platform management console unavailable 259
- Unwired Server
 - applying configuration changes 58
 - checking status 44
 - configuration changes unsuccessful 268
 - extended session 256
 - importing package contents 190
 - improving BES MDS HTTP traffic
 - performance 55
 - list does not appear in SCC 263
 - logging 67
 - logging out of 13
 - moving package contents from or to 190
 - pinging 44
 - refresh after changing configuration 268
 - server list 43
 - services provided by 42
 - startup failure 265
 - stopping and starting 43
 - suspending and resuming 44
- Unwired Server configuration
 - refresh 57
- Unwired Server properties 59
- Unwired Servers
 - administration overview 4
- user interface, about 18
- users 173
 - able to connect with old password 269
 - administering 172
 - administration overview 5
 - administration, configuring 78, 133
 - administration, maintaining 78, 133
 - deleting 172

Index

- devices used by 173
- messaging statistics 245
- mobile workflow, checking 217
- monitoring 244
- not displayed for registered devices 273
- security statistics 230

V

- variables, context
 - configuring 214
- view layouts
 - cascade 22
 - horizontal tiling 22
 - vertical tiling 22
- Viewing applications 170
- views
 - about 21

- bringing to front of perspective 21
- closing 21
- icons for managing 18
- maximizing 21
- minimizing 21
- opening 21
- restoring 21

W

- Windows
 - starting, stopping Sybase Control Center 10
- workflows, mobile
 - See mobile workflows