



Troubleshooting

Sybase Unwired Platform 2.1

ESD #3

DOCUMENT ID: DC01206-01-0213-02

LAST REVISED: January 2013

Copyright © 2013 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.

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 Oracle and/or its affiliates 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

Introduction to Troubleshooting	1
Documentation Roadmap for Unwired Platform	3
Troubleshooting Basics	5
Collecting Information	5
Log Files	5
Mobile Application Project and Deployment Information	11
Viewing Messages in Sybase Control Center	12
Reporting Errors	13
Reproducing Problems	13
Sybase Technical Support Checklist	14
Sending Error Log Fragments	15
Ensuring System Recoverability	16
Troubleshoot Basic Scenarios	17
Sybase Unwired Platform End-to-End Process Flow	17
Debugging Message Synchronization	20
Debugging Replication Synchronization	23
Debugging the Cache	26
Debugging Data Change Notification	28
Debugging Android Device Development	32
Debugging BlackBerry Device Development	35
Debugging iOS Device Development	37
Debugging Windows and Windows Mobile Object API Development	41
Debugging DOE-C End-to-End Process Flow	44
Debugging JCO End-to-End Process Flow	46
Troubleshoot the System	49
Troubleshoot the Postinstallation Configuration	49
Troubleshooting Host Name Problems	49
Listeners or Services Could Not be Started Errors	50

Sybase Unwired Server Service Fails to Restart After Credential Change	56
Handling Intrusion Detection/Prevention Software	56
Cannot Access Device Tab and Web Service Error	57
Cannot Access sampledb	58
System Environment Path is Too Long	58
[Deprecated in 2.3]Client Application Hangs Waiting for Authentication	58
Performance Issues if Unwired Server Not Installed on a 64-bit Machine	60
Unwired Server or RSOE Startup Problems	60
.NET Environment Problems Affect Unwired Server	62
Messaging Installer Completed With Errors	62
Troubleshoot Deployment	63
Result Set Filter or Result Checker Class Files Not Deployed To Unwired Server	63
Modified Configuration Profile Changes Not Deployed	63
NOCLOBBER Deployment Error	64
Troubleshoot Synchronization	65
Synchronization Failed while Downloading Data	65
Multi-threaded Delete Failures Not Logged	65
Device Synchronization Returns no Data	65
Errors When Reusing Primary Keys	66
Troubleshoot Runtime	66
Errors if CUD operations are not defined for an MBO	66
Troubleshoot the Unwired Server Cache	67
Primary Key Record Missing	67
Data Caching Problems	67
No Data or Images Appear on the Device	68

Troubleshoot Security, Logins, and Authentication	68
Cannot Log in to Unwired Server	68
Troubleshooting Single Sign-On	69
Server Key Verification Failed	70
Troubleshoot EIS Connections	70
DOE-C or Messaging Connection Test Errors	71
Data Source Performance Errors	71
SAP Issues	72
Troubleshoot Servers	73
Unwired Server	73
Relay Server	81
Troubleshoot Clusters	83
Advantage Database Server Cannot Authenticate	83
Nodes in Cluster Do Not Start Because of JVM Heap Space	84
Messages from Shutting Down the Primary Server in a Cluster Environment	84
Database Services Do Not Start Automatically ...	85
Unwired Server Cannot Start in Domain Version Mismatch	85
Troubleshoot Workflow DCN Failures	85
No Workflow Message is Sent	86
Troubleshoot the Sybase Control Center	87
Using Sybase Control Center to Troubleshoot Unwired Platform	87
Collecting Administration Performance Data for Troubleshooting	88
Sybase Control Center Management Tier Issues	89
Launching SCC Results in Rounded Rectangle Box or Empty Console Screen	89
Sybase Control Center Windows Service Fails to Start	90
Sybase Control Center Windows Service Deleted	91

Sybase Control Center Fails to Start	91
Second Sybase Control Center Fails to Start	91
Login Invalid in Sybase Control Center	92
Administrator Account is Locked	92
Browser Refresh (F5) Causes Logout	93
Stale Version of Sybase Control Center After Upgrade	93
Sybase Control Center Reports Certificate Problem	93
Previous Administrator Credentials Used	94
Security Error Triggered When Connecting to SCC from Remote Browser	94
Administrator Login Passes When Provider Is Not Available	94
Host Name of Registered Resource Changed But Is Not Updated	94
Management Issues with Clustered Data Tiers ...	95
Poor Sybase Control Center Performance after Upgrade	95
Sybase Control Center Communication with Unwired Server Fails	95
Platform Component Monitoring Issues	96
Monitoring Data Does Not Appear in History Tab	96
Domain Log Data Does Not Appear in History Tab	97
Previously Existing Monitoring Data No Longer Appears	98
Previously Existing Domain Log Data No Longer Appears	98
Server Tier Administration Issues	98
Server List Not Retrieved	99
Unwired Server Fails to Start	100

Error in Listing Application Connections and
ADMIN_WEBSERVICE_INVOCATION_ERR
OR in gateway.log101

Starting or Restarting a Remote Server from
Sybase Control Center Fails101

Port Conflict Issues103

Unexpected Listener Startup or Connection
Errors103

Refreshing Server Configuration Displays Only
Partial Updates103

Users Connect with Old Credentials104

AuthorizationException Displays Instead of
Status 105

Increasing Messaging Queue Counts Degrades
Performance 105

Saving Server Configuration Fails Due to
Certificate Validation Error105

Unknown Server Error Message106

Package Deployment and Management Issues 106

Exporting or Deploying Large Packages Fails ... 106

Invalid DOE-C User Error for an SAP Server
Connection 107

Application and Application User Management Issues
..... 107

Wrong Application for Code Error107

User Name of Registered Application
Connection Not Displayed108

Internal Server Error When Clicking Applications
..... 109

Troubleshoot Mobile Business Object Development111

Unwired WorkSpace Crashes When Previewing or
Test Executing Large Amounts of Data 111

Cannot Start Unwired WorkSpace After Modifying
Plugins111

Cannot Access Sample Database After Switching Network Types	112
Data Source Connection Fails After Installing to a New Directory	112
JDBC Connection Errors	112
Mobile Business Object Errors	113
Binding an MBO to a Database Fails	113
Multilevel Insert Operation Fails	113
MBO Retrieves Incorrect Data	113
Relationship Error	114
Warning Message When Creating Relationships	114
Troubleshooting Default Values for Adaptive Server [®] Anywhere uniqueidentifiers	114
AutoCommit Option for JDBC MBOs Using an ASE Data Source	115
Passing an Empty String or Byte into a Required Binary Column in ASE	116
Preview Errors for a Web Service Mobile Business Object with date/time Types	116
Web Service Mobile Business Object Must Be Connected to Connection Profile	117
Refresh Fails Due to Missing Load Argument Default Values	117
Input Parameters Fail to Generate for Complex Data Sources	117
Output Mapping Warning Message for Multilevel Insert Operations	117
Attribute Length Set Incorrectly for Operation and Load Arguments	118
Unable to properly update the CDB because the partition was improperly assigned	118
Syntax Errors When Creating a Database Mobile Business Object	119
Sybase MBOs That Invalidate the Cache	119

Deployment Errors	119
Deploying to Unwired Server Fails	119
Deploying to Unwired Server in a Cluster Fails .	120
Deploying a Renamed MBO Fails	120
Troubleshoot Code Generation	120
Code Generation Fails	120
SQL_MAX_ROW_SIZE_EXCEEDED	121
Namespace Name Does Not Exist Error	121
Cannot Generate Code with Javadoc Option	
Enabled	122
Installing the net_rim_api.jar or android.jar File .	122
Error Messages About Missing Directories	
When Generating Object API Code	122
Troubleshoot Unwired WorkSpace Errors	123
Unwired WorkSpace Does Not Support Saving	
Files as PDF	123
Installing Unwired WorkSpace Into a Different	
Directory	123
Avoiding Out of Memory Errors During Preview	
and Test Execute	123
OSGI Bundle Related Error Messages	123
Unwired WorkSpace Crashes and Fails to	
Restart	124
Troubleshoot Object API Development	125
Debugging Replication Synchronization	125
Insert Operation Fails for iOS	127
MBO Update Fails	128
Windows Mobile Compile Error For Generated API	
Project	129
Cannot Insert DEFAULT in Timestamp Column	129
Synchronization to SAP Data Source Repeats	
Excessively	129
Error When Running Generated 32-bit Windows Code	
on a 64-bit Machine	130

Error with Multiple Writer Threads Modifying the Same Table Row	130
Relaunched Application Gets Encryption Key Error ...	131
Encryption Key Cannot Contain Special Characters . .	131
MinCheckAffectedRowCount behavior with JDBC endpoints	131
Problems with a Slow Query	132
Troubleshoot Hybrid Web Container Development	133
Debugging Message Synchronization	133
Debugging Hybrid Web Container Development	136
Debugging Hybrid Web Container Connection Issues	137
Troubleshoot Hybrid App Development	138
Exception Received with Large Amounts of Data	138
Hybrid App Is Not in the Client List	139
Hybrid App Does Not Receive Notifications	140
Settings Program Does Not Appear on Windows Mobile Device	141
Failed to Create Second Process	141
Changes to Custom.js File Do Not Work	141
Hybrid App Package Deployment	142
Operation Fails with Login Failed Error	142
Operation Fails with Package not Found Error . .	143
Operation Fails with NULL Values Not Supported Error	143
Object Query Listview Does Not Display Data . .	143
MincheckAffectedRowCountException Error Received	144
Hybrid App Push Notification Not Processed ...	144
PhoneGap API Framework Not Invoked on Device	145
Hybrid App Stops Working Properly after Upgrade	145
Troubleshoot REST API Development	147

Cannot Access External Services	147
Application Reports a 401 Unauthorized Error	147
Security Configuration Users Not Allowed Domain Access	147
Proxy Connection Not Found	148
Troubleshoot OData Development	149
Application Reports a 403 Forbidden Error	149
Unable to Delete NoSecLoginModule From the New Security Configuration	149
Native Push Notification Not Working After ODP Upgrade	150
Application Reports a 401 Unauthorized Error	150
Troubleshoot Agentry Development	151
Troubleshoot Device Clients	153
Device Provisioning	153
Device Cannot Communicate with Afaria Server	153
Problem Loading .cod Files on the BlackBerry Device	153
Device Clients	153
No Workflow Message is Sent	153
Synchronization Fails in a Remote Server Environment	154
Synchronization Fails With OutOfMemory Error	154
Synchronization Fails With Exception	155
Timeout Errors During Client Synchronization ..	155
Device Operation Fails With SQLE_NOT_IMPLEMENTED Error	156
No Data Displays on Device Client	157
Synchronizing the Device Application Returns Only One Row of Data	157
Packet Dropped State	157
Pattern is Too Long Exception	159
Feature Not Implemented Exception	160

Synchronization Key Rows Must be Unique	160
BlackBerry	160
iOS	164
Windows Mobile and Windows	165
Android	168
Troubleshoot Command Line Utilities	169
Utilities Cannot Locate djc-setenv.bat	169
Validation Rules	171
Mobile Business Object Validation Rules and Error Messages	171
Mobile Business Object, Attribute, and Structure Object Error Messages and Validation Rules	172
Parameter and Argument Validation Rules and Error Messages	179
Operation Validation Rules and Error Messages	188
Relationship Validation Rules and Error Messages	195
Cache Validation Rules and Error Messages	207
Synchronization Validation Rules and Error Messages	209
Object Query Validation Rules and Error Messages	212
Personalization Key Validation Rules and Error Messages	214
Custom Filters, Roles, and Synchronization Group Validation Rules and Error Messages	216
Mobile Workflow Validation Rules and Error Messages	218
Glossary: Sybase Unwired Platform	227
Index	243

Introduction to Troubleshooting

This troubleshooting guide provides information about resolving configuration, development, deployment, and production problems that you may encounter using Sybase® Unwired Platform.

The troubleshooting guide includes a collection of symptoms, with information to further diagnose or solve the problem. These presenting symptoms are organized into categories. You can either check the appropriate category for a problem, or you can search the *Troubleshooting* online help for the symptom you have experienced, or for an error message you have observed.

Troubleshooting categories include:

- Basic scenarios
- Sybase Unwired Platform system
- Sybase Control Center for Sybase Unwired Platform
- Mobile business object development
- Device application development
- Device clients
- Validation rules for mobile business objects processing

Use the troubleshooting guide with other Sybase Unwired Platform documents.

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.

Documentation Roadmap for Unwired Platform

Troubleshooting Basics

Sybase Unwired Platform provides tools for gathering information about system health properties. The Sybase Control Center administrative console, debugging, and tracing, as well as native operating system commands, and Unwired Platform command line tools combined offer root-cause diagnosis and problem resolution abilities to development and operational staff working with Sybase Unwired Platform.

The primary goals of troubleshooting:

- Gather enough information to diagnose the problem, so you can solve it.
- Gather supporting information to document the problem, so you can contact Sybase Technical Support for help in resolving it.

Collecting Information

Use Sybase Unwired Platform and other resources to collect information, then use the information to troubleshoot and solve the problem.

Sybase recommends that you develop a baseline for Sybase Unwired Platform system operations when things are running smoothly. This provides a benchmark for comparison when you encounter problems, and can lead to ideas for where to focus your troubleshooting efforts.

Sybase Unwired Platform resources include the Sybase Control Center Web console, error messages, log files, monitoring facilities, and the trace feature. Additional system resources include the system event log, operating system monitoring utilities, and third-party utilities.

The Visual Studio debugger is also a good resource for Windows and Windows Mobile device applications; use it to walk through code on the client user interface side.

Another resource is the Solved Cases database on the Support and Services Web site: <http://www.sybase.com/support>.

Log Files

Unwired Platform log files provide a source of troubleshooting information. Log file information is aggregated and displayed in the Sybase Control Center administrative console.

Accessing server logs

The server log records all system events as per the log level configuration for individual sub-components. For detailed diagnostics, you can configure individual sub-system's log level to DEBUG or TRACE level, and the messages recorded in the server log can help identify the problem. It is not recommended to leave sub-system log levels to those considering the

Troubleshooting Basics

performance impact and growth of the log file when large number of users may be accessing the system. Server log is separate for each server.

Server log settings are controlled from Sybase Control Center. Server log configuration settings changes on the primary server are propagated to other nodes within one minute, and do not require a cluster changing action.

In Sybase Control Center, navigate to *Servers > ServerName > Log node > General tab*.

You can also view the server log from the command line using a text editor or any other third-party log viewing tool. (Sybase Control Center helps to view the server log remotely without needing access to the server's file system).

- The main server log file is `SUP_HOME\Servers\UnwiredServer\logs\hostname-server.log`.
- The incoming HTTP requests log (for Data Change Notification, and others) is in the `hostname >-http.log` file. You may also see some additional folders which contain log files produced by the internal modules of the Messaging Server. The key folders are `JMSBridge` and `SUPBridge`. The messages recorded in those files are controlled by the log level setting of the `MSG` sub-component of the Unwired Server logging system.

These key sub-systems can help troubleshoot issues:

- **Data Services** – consider modifying this sub-system log level if you are trying to diagnose issues with backend EIS interaction, caching, and Data Change Notification.
- **MMS** – consider modifying this sub-system log level if you are trying to diagnose issues with client interaction, message-based synchronization application activities such as subscription command processing, data import messages, etc., and replication synchronization application activities such as those during upload and download phases of the synchronization, data change notification processing, background tasks such as cache refresh, etc. among others.
- **MSG** – consider modifying this sub-system if you are need to look at the individual messages being exchanged between one or more messaging synchronization application clients and server.

Accessing domain logs

The domain log shows data only if domain logging is enabled for that domain, and one of the packages in the domain has package-level logging enabled. Domain log data is automatically purged based on the domain log setting for the domain. Domain log is a cluster-wide logical view of the activities and errors.

Navigate to *Domains > DomainName > Log node > General tab*.

Accessing client logs

If client logging is enabled through the client object API, the client logs provide critical information about the application running on the device and its operational state. The client application information is uploaded from the device to Unwired Server, where it is visible using Sybase Control Center.

In the left navigation pane, expand the Packages folder and select the package you want to investigate, then in the right pane, select the **Client Log** tab.

Server Logs

Server logs provide a view of Unwired Server activity. Status and error messages are collected. You can set the level of granularity for messages captured, and the log file size through Sybase Control Center. In a troubleshooting situation, you may want to increase the granularity of messages captured.

Search *Sybase Control Center for Sybase Unwired Platform* online help for the topic *Configuring Server Log Settings* for information.

Windows Event Log

Sybase Unwired Platform system messages are logged in the Microsoft Windows application event log.

Events generated by respective platform components are logged with the following source identifier values.

- Sybase Messaging Server
- SQLANY 12.0 (32 bit) or SQLANY64 12.0 (64-bit)
- Unwired Server
- Sybase Control Center *X.X*

The following events are logged.:

- Server start and stop events are recorded as information events.
- Server errors such as license errors, failure to connect to runtime databases, and so forth, are logged as error events.
- Whenever a user account is locked due to repeated login failures, a message is logged as warning event.

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.

Client Logs

Client logs capture data about client application operations for all devices subscribed to a package. You can use the data to track errors and identify performance issues.

By default, logged data includes the user name, device ID, mobile business object with which the client is syncing, the operation the client is performing, the success or failure of the server-

Troubleshooting Basics

side operation, current log level, date and time stamp, the log message associated with the message.

Developers use client-side APIs to enable this capability to log critical application data on the device that can be uploaded automatically to the server where it is visible in the package's client log. The capability can be implemented so that enabling and disabling logging does not require any user intervention, is transparent, and can be turned on or off automatically.

See *Sybase Control Center > Manage > Managing Unwired Platform > Routine System Maintenance Tasks > Checking Client Application Logs*.

Unwired WorkSpace Log

You can view the Unwired WorkSpace log in the Error Log view, to see error messages from Unwired WorkSpace, exceptions, and stack traces. When gathering information to diagnose a problem, first collect the Unwired WorkSpace log, and review that information in combination with necessary server log entries (if needed) to diagnose the root cause.

Log File Locations

Use a text editor to review log files from the command line using a text editor if Sybase Control Center is not available, or to concentrate your review to a particular log.

Table 1. Log file locations

Log type	Location
Unwired Server	<p>Aggregated Unwired Server logs: <i><UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\UnwiredServer\logs</i></p> <p>Unwired server log: <i><UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\UnwiredServer\logs\<host>_server.log</i></p> <p>Database error log: <i><UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\UnwiredServer\logs\errorlog.txt</i></p> <p>MobiLink Server error log: <i><UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\UnwiredServer\logs\mlsrv_err.log</i></p> <p>Bootstrap log: <i><UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\UnwiredServer\logs\bootstrap*.log</i></p> <p>Messaging service log details: <i><UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\UnwiredServer\logs\<Module></i></p> <p>Device client tracing logs: <i><UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\UnwiredServer\logs\ClientTrace</i></p> <p>Mobile Workflow tracing logs: <i><UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\UnwiredServer\logs\WorkflowClient</i></p>

Log type	Location
<p>Sybase Control Center for Sybase Unwired Platform</p>	<p>Sybase Control Center agent log: <UnwiredPlatform_InstallDir>\SCC-XX\log\agent.log</p> <p>Gateway log: <UnwiredPlatform_InstallDir>\SCC-XX\log\gateway.log</p> <p>Repository log: <UnwiredPlatform_InstallDir>\SCC-XX\log\repository.log</p> <p>Request logs: <UnwiredPlatform_InstallDir>\SCC-XX\services\EmbeddedWebContainer\log\request-<yyyy_mm_dd>.log</p> <p>Database server log: <UnwiredPlatform_InstallDir>\SCC-XX\services\Repository\scc_repository.slg</p>
<p>Relay server and RSOE</p>	<p>Default log details:</p> <p>%temp%\ias_relay_server_host.log</p> <p>%temp% is the Windows environment variable, for example, C:\WINDOWS\Temp.</p> <hr/> <p>Note: In the case of IIS Web servers, the log file is C:\WINDOWS\system32\LogFiles. However, this location can be configurable in IIS Manager.</p> <hr/> <p>RSOE log files, by default:</p> <p><UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\UnwiredServer\logs\<nodeName>.RSOE<n>.log</p> <p>For example: RBSHost1.RSOE4.log.</p>
<p>Installer</p>	<p><UnwiredPlatform_InstallDir></p> <p><UnwiredPlatform_InstallDir>\UnwiredPlatform\InstallLogs</p> <p><UnwiredPlatform_InstallDir>\UnwiredPlatform\InstallLogs\silentInstall</p>
<p>ESD Installation Log</p>	<p><UnwiredPlatform_InstallDir>\UnwiredPlatform\supXXebfxlogs</p>
<p>Domain</p>	<p>In the domainlog database; viewable in Sybase Control Center.</p>

Log type	Location
Cache database logs	<p>By default, cache database errors are logged in the <code>error-log.txt</code> file, located in <code><UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\UnwiredServer\logs</code>.</p> <p>For a data tier installation, there are three database server logs, by default located in <code><UnwiredPlatform_InstallDir>\UnwiredPlatform\Data\CDB</code> (but may vary depending on the installation):</p> <ul style="list-style-type: none"> • <code>errorlog.txt</code> – cache DB server log • <code>clusterdb_errorlog.txt</code> – cluster DB server log • <code>monitordb_errorlog.txt</code> – monitor and domainlog DB server log
Unwired WorkSpace log	<p>In Unwired WorkSpace, use the Windows > Show View > Other > Error Log view, or collect the log file from the Eclipse workspace directory at <code><workspace folder>/.metadata/.log file</code>.</p>

Mobile Application Project and Deployment Information

The Mobile Application Project and the associated deployment unit file provide a source of troubleshooting information when there are MBOs involved; for example, MBOs are involved when troubleshooting a device client (including object API clients (all platforms), or mobile workflow clients that use MBOs), or when debugging cache database or DCN issues.

Collect this information on the mobile application project:

- The Mobile Application Project (created in Unwired WorkSpace)
 - In WorkSpace Navigator, right-click on the mobile application project folder, and select 'Export' from the context menu to invoke the Export wizard.
 - In the 'Select' wizard page, select **General > Archive File**, and click **Next**.
 - In Archive File wizard page, click **Browse** and provide the zip file path in the dialog.
 - Click **Finish**.
- The `deployment_unit.xml` and `deployment_descriptor.xml` files.

For information on how to save the deployment profile which contains these files, see *Sybase Unwired WorkSpace - Mobile Business Object Development > Develop > Deploying a Mobile Application Project*.

Once you have saved the deployment profile, you can extract the `deployment_unit.xml` and `deployment_descriptor.xml` files from the zip file in Unwired WorkSpace at: `<WorkspacePath>\<Mobile Application Project Name>\< Deployment >\ <Mobile Application Project Name>.profile > \<Unwired Server name selected during deployment> \ <Mobile Application Project>.zip`.

Viewing Messages in Sybase Control Center

Sybase Control Center for Sybase Unwired Platform can be used to review an ongoing stream of system messages to keep you alert to system health and processing status.

Use the Sybase Control Center administrative console to view system and error messages on an ongoing basis. See the *Sybase Control Center for Sybase Unwired Platform* online documentation for more information about using the console to view messages and log files.

Error Message Severity Levels

Error message severity levels provide an indication of system health, and how seriously to take a message.

Do not view individual messages in isolation, however, since a system event may trigger a range of symptoms that individually may not seem related. Review messages as a group, look for trends, and view upstream messages in the log for possible initiating events.

Table 2. Error message severity levels

Severity level	Description
Trace	Finer-grained information than Debug.
Debug	Fine-grained informational events that are useful for debugging an application.
Info	Informational messages that highlight application progress at a coarse-grained level.
Warn	Warning messages that indicate a potentially harmful situation.
Error	Error events that may allow the application to continue running.
Console (off)	Messages sent to the <code>system.out</code> stream logs for the Sybase Control Center administrative console.

Variables in Error Message Text

Unwired Platform error messages are frequently composed of one or more variables. Unwired Platform components, such as Unwired Server, and back-end enterprise information systems (EIS) replace the variable with the appropriate system information.

This informational message includes several variables:

```
INFO: SUCCESS:<Userid>:<MBO name>[.Operation name]:<effective parameter-value pairs>: [Warning message from EIS]
```

This warning message includes several variables:

```
WARN: FAILED:<Userid>:<MBO name>[.Operation name]:<effective parameter-value pairs>: <Error message from EIS>
```

Variables are shown in brackets:

- <Userid> – a user id will be inserted in the error message.
- <MBO name>[.Operation name]:<effective parameter-value pairs> – a string of variable information will be inserted, including the mobile business object name, an operation name, and parameter-value pairs.
- <Warning/Error message from EIS> – a warning or error message from the enterprise information system (EIS) will be inserted.

Reporting Errors

Before reporting problems to Sybase Technical Support, collect as much relevant background information as you can.

For application related issues, the relevant information would be the original Eclipse project that contains the mobile business objects and the client project, server log files, error messages, symptoms, observations, and steps you have taken to solve the problem.

For server administration issues, relevant information would be in the server log files and Sybase Control Center log files.

This information may help the technician reproduce and diagnose the problem.

Reproducing Problems

A problem that can be reproduced may help your technician more quickly solve the problem. "Reproducible" means that the code causing the incorrect behavior can be isolated, and that the issue can be demonstrated. It may also be useful to narrow the reproduction by removing extraneous steps or code.

Provide as much useful information as possible on your problem report, including a reproducible case.

Sybase Technical Support Checklist

Before reporting errors, collect information about Sybase Unwired Platform components.

Component	Information
Unwired Server, clients, applications, devices, users	<ul style="list-style-type: none"> • Unwired Server version and build; the version information is available in the server log as well displayed in <i>System Administration</i>. • Full description of incorrect behavior, including all components involved • Server and client errors (stream error) and error messages (Unwired Server log files) <hr/> <p>Note: See <i>Unwired Server Runtime Logging in System Administration</i>.</p> <hr/> <ul style="list-style-type: none"> • Client device platform, version, and build (if appropriate) • Application information (if appropriate) • User name and identifier information • Cache database error log file, which by default is <code>error-log.txt</code> in <code>SUP_HOME\Servers\Unwired-Server</code>. • For administration console issues, include <code>agent.log</code>, and <code>gateway.log</code> files. The Web browser type and Adobe® Flash® Player version information should also be included.

Component	Information
Relay server, domains, clusters [separate installation]	<ul style="list-style-type: none"> • Relay server version and build • Full description of incorrect behavior, including all components involved • Relay server-related errors and error messages <hr/> <p>Note: See <i>Configuring and Enabling Relay Server Logging in System Administration</i>.</p> <hr/> <ul style="list-style-type: none"> • Type of Web server (IIS or Apache), version, and operating system • Number of relay servers • Indicate whether the relay servers are in a farm, and provide configuration information <p>If a load balancer is used:</p> <ul style="list-style-type: none"> • Indicate whether a hardware load balancer or software load balancer is used • Number of Relay Server Outbound Enablers (RSOEs) configured for each Unwired Server. • Indicate whether the RSOEs are configured with HTTP or HTTPS to relay server. • RSOE log error and error messages <hr/> <p>Note: See <i>Configuring RSOE Logging in System Administration</i>.</p> <hr/>
Afaria® [separate installation]	<ul style="list-style-type: none"> • Afaria version and build • Full description of incorrect behavior, including all components involved • Afaria-related errors and error messages

Sending Error Log Fragments

Error logs are a valuable source of information for troubleshooting problems. Provide error logs, or error log fragments, when reporting your problem. Also provide a timestamp of when the problem occurred, so that logs can be compared.

In many cases it is advisable to back up the log file, clear its contents, reproduce the problem, and then send log file. This reduces the content of the log file to pertinent information needed to diagnose the problem.

If possible, remove or un-install un-necessary packages from Unwired Server, then reproduce the problem. When reproducing the problem, try, as much as possible, to omit unnecessary steps. If you have cleared the log, as described above, before reproducing the problem, only relevant information will be logged. Another good idea for capturing diagnostic information is

Troubleshooting Basics

to take error screen shots, or record videos that capture reproducible steps. These practices significantly reduce unnecessary information in server logs and help the troubleshooting process.

To prepare error log file fragments:

1. Identify the log files that may include important information.
2. Copy error log files to a separate directory, or copy a section from an error log file to a separate text file.
3. Include the error log file or files, as well as screen shots and video logs, with the problem report you submit to Technical Support.

Ensuring System Recoverability

Ensure system recoverability by keeping Sybase Unwired Platform components in good health, understanding your topology, and implementing sound operations and maintenance practices. Develop a disaster recovery plan that includes backing up the installation directory, cache database (CDB), and transaction logs.

See *System Administration* for information about maintaining system health, and backing up and restoring Sybase Unwired Platform.

Troubleshoot Basic Scenarios

Troubleshooting information for debugging process flow for the system, and for device application development; identifies useful tools and utilities.

Sybase Unwired Platform End-to-End Process Flow

Understanding Sybase Unwired Platform processing flow at a high level between the enterprise information system (EIS), Sybase Unwired Platform, and devices is important for troubleshooting, as well as knowing the options available for diagnosing problems at each step of the process.

High-level process flow

This diagram shows a logical view of Sybase Unwired Platform communications between the EIS, Sybase Unwired Platform components, and device clients. The diagram also shows relay server and relay server outbound enabler (RSOE), which are optional components.

Figure 1: Sybase Unwired Platform communications

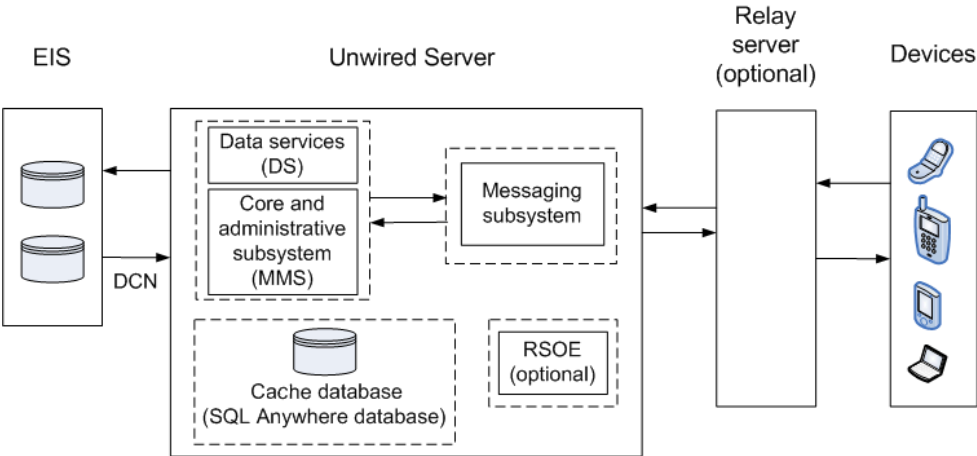
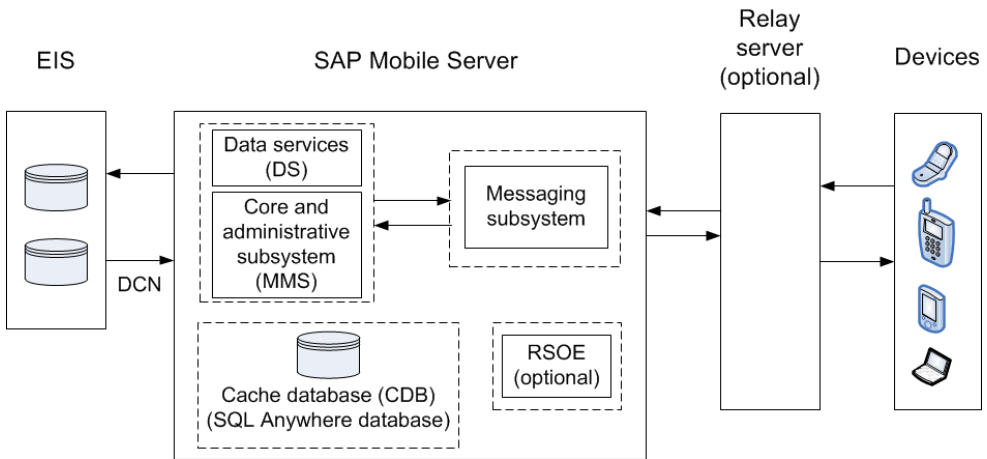


Figure 2: Sybase Unwired Platform communications



Unwired Server subcomponents include:

- Core and administrative subsystem (MMS) – responsible for core services which include device client application interaction management, security enforcement, notifications processing, and runtime component management. MMS administrative services provide the server management interface used by Sybase Control Center.
- Data services (DS) – responsible for back-end interaction, data integrity, and transactions.
- Cache database (CDB) – is a runtime cache database used by server-name. By default, an embedded SQL Anywhere® database server is used as the CDB, as shown in the diagram. However, you can configure server-name to use an existing SQL Anywhere instance as its CDB server.
- Messaging subsystem – responsible for transport of messages required for maintaining data integrity on device client applications. It uses a JMS service to upload and download data changes between devices and the cache database.

Troubleshooting options

This section describes the options available for diagnosing process flow problems. Details are provided separately.

Table 3. Options for diagnosing process flow

Processing component	Diagnostic option
Enterprise Information System (EIS)	Use EIS administrative tools to make sure the data source is available, and check logs for possible communications problems between the EIS and Unwired Server.

Processing component	Diagnostic option
Unwired Server – data services (DS)	<ul style="list-style-type: none"> • Using Sybase Control Center, enable data services subsystem logging. • Review interactions for getting data and sending transactions.
Unwired Server – core and administrative subsystem (MMS)	<ul style="list-style-type: none"> • Using Sybase Control Center, enable core and administrative subsystem logging. • Monitor packages. • Enable package-level logging. • Enable synchronization tracing. • View the mobile business object (MBO) operation history.
Unwired Server – messaging subsystem	<ul style="list-style-type: none"> • Using Sybase Control Center, enable messaging subsystem logging. • Using Sybase Control Center, enable messaging server-side logging for one or more of its modules. Note, the messaging subsystem logging controls three modules (SUPBridge, JMSBridge, and WorkflowClient), but other modules may be useful to debug a problem if recommended. • Retrieve client-side messaging-based synchronization trace by sending a request to the messaging device from Sybase Control Center. • Check the module log for SUPBridge and JMSBridge modules messages. The SUPBridge log contains incoming messages from device clients, and the JMSBridge log contains messages going to device clients.
Relay server and Outbound Enablers (RSOE) – if used	<ul style="list-style-type: none"> • Check the relay server configuration. • Enable relay server tracing (in <code>rs.config</code> on the relay server machine), and outbound enabler log verbosity using Sybase Control Center. • Review relay server and outbound enabler log files. • Check Microsoft Windows system log files, and web server (IIS or Apache) log files.

Processing component	Diagnostic option
Device clients	<ul style="list-style-type: none"> • If the SUPLogger is implemented as part of custom coding, enable Show Log, and set the trace flags. • Enable WorkflowClient tracing from the client. • Use Sybase Control Center to check device information and settings, including the current connection status. • Review log files uploaded from the device. • Check the configuration, especially connections between the device and Unwired Server, and the certificate. <p>If the device is offline (or cannot connect to Unwired Server), you may not be able to retrieve device logs (for example, messaging logs) from the device through Sybase Control Center. If you have physical access to the device, you may connect the device to your computer (either PC or Mac), and use certain third party software to retrieve those log files. For example, you can use iPhoneExplorer to browse through iPhone to extract logs; Windows Explorer to extract logs from Windows Mobile; and Android AVD Manager to get logs from Android.</p>

Debugging Message Synchronization

Describes the message synchronization process flow, and options available for diagnosing problems at each step of the process.

Process flow

This diagram shows a logical view of message-based synchronization communications between the enterprise information system (EIS), Sybase Unwired Platform components, and device clients. Relay server is included as an optional component.

Figure 3: Message synchronization process flow

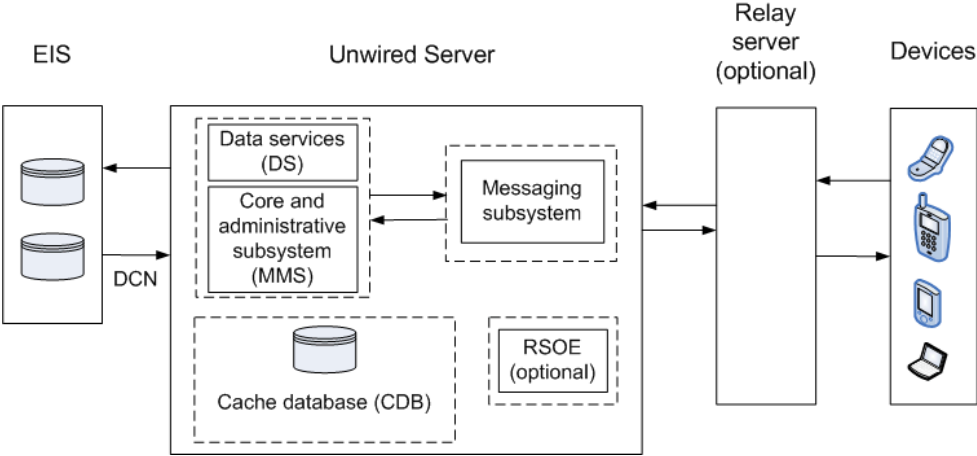
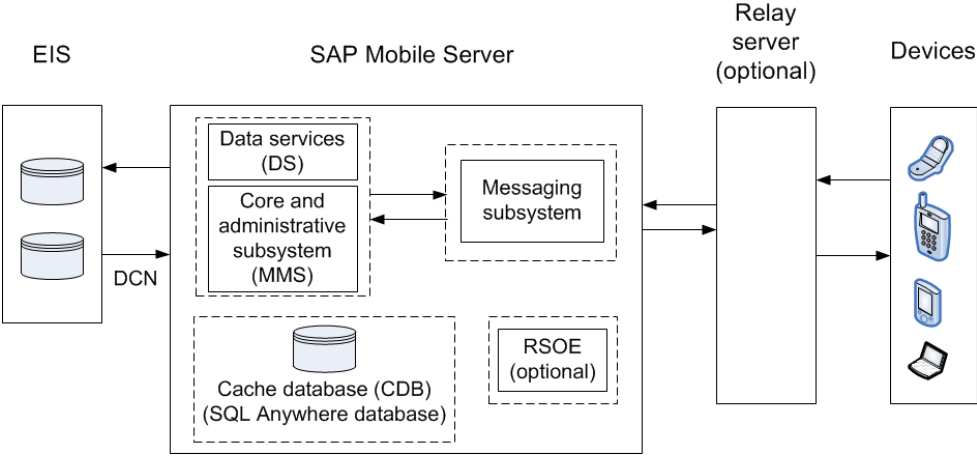


Figure 4: Message synchronization process flow



Troubleshooting options

This section describes the options available for diagnosing message synchronization process flow problems.

Unwired Server – data services (DS):

- In Sybase Control Center, enable data services (DS) subsystem logging by setting its log level. The output is written in the server log.
- Review subsystem messages for information about getting data and sending transactions.
- For cache activities, you can enable domain logging on the package using Sybase Control Center, and then review data in cache sub-systems (Cache Refresh and DS Interface

specifically) for the period domain logging is enabled. See *Domain Logs* and *Domain Log Categories* related documentation in *System Administration* and *Sybase Control Center for Sybase Unwired Platform* online help for complete details.

Unwired Server – core and administrative subsystem (MMS):

- If you like to troubleshoot at the system-level (which will record all domains and packages activities), you can enable MMS logging by setting the log level in Sybase Control Center. The output is written in the server log.
- Select the desired application package, and enable domain logging on that package. Review the captured log messages in Sybase Control Center.
- For quick troubleshooting to locate system errors, you can check the package-level client log and MBO/Operation history in the Sybase Control Center.
- View the MBO/Operation history in Sybase Control Center for errors during back-end interaction with the EIS.
- On the data-tier node, review the database error logs:
 - `errorlog.txt` – database system errors
 - `clusterdb_errorlog.txt` – cluster DB server log
 - `monitordb_errorlog.txt` – monitor and domainlog DB server log

Unwired Server – messaging subsystem :

- Use Sybase Control Center to enable server-side logging for <MSG> log subcomponents.
- Retrieve client-side message-based synchronization trace by sending a request to the MBS device using Sybase Control Center (on the **Application Connections** tab, use **Get trace**). A client's log level can be set in the device settings of the Sybase Control Center.
- Configure messaging server-side module level tracing (for users with advanced knowledge).
 - Use Sybase Control Center to set the log levels of individual modules. Select **Servers** > <server_name> > **Log** > **Messaging Server** > **Settings** > <module> > **Properties**.
 - If debugging a Hybrid App, turn on WorkflowClient tracing.
 - Check the module logs – SUPBridge and JMSBridge modules, or the WorkflowClient module (to debug Hybrid Apps).
- On Unwired Server, review the `errorlog.txt` for any cache database system errors: `SUP_HOME\Servers\UnwiredServer\logs\errorlog.txt`.

Relay server – if used:

- Enable relay server tracing by setting verbosity to 1 or higher. You can do that by editing the `rs.config` file of your Relay Server and running the `rshost` utility. Check the *System Administration* topic on Relay Server Setup.
- Review the relay server log files. The logging for `rshost.exe` is directed to the `myrshost.log` file in the `C:\temp` directory.
- Check the IIS log files (`windows\logfiles\W3SVC1` and `HTTPERR`). The IIS server log path is configurable from `inetmgr` from **Default Web Site** > **Properties** > **Website tab**

> **Enable Logging** > **Properties** > **Log file directory**. The recommended location is C : \Inetpub\LogFiles.

On Unwired Server, check the Sybase Unwired Platform outbound enabler log for replication synchronization entries: `SUP_HOME\Servers\UnwiredServer\Logs\NodeName.RSOEnum.log` (for example, `smpqa01SMPServer1a.RSOE1.log`).

Device clients:

- For Hybrid Apps running on Android, BlackBerry, and Windows Mobile, run the **Settings** application, and choose **Show Log**. The client system log for messaging activities appears. Key information includes messages related to server connectivity, server messages, and messaging client requests. Native applications may or may not include **Settings** and **Show Log**, depending on the design.
- Set the device log trace level on the device and get the device logs from Sybase Control Center by selecting **Applications** > **Application Connections** > <device> > **Properties** > **Device Advanced** > and setting the **Debug Trace Level** to 4 or higher.
After running the application on a device, select the device in Sybase Control Center and select **Get Trace**. The client trace log on the device is transferred to the Unwired Server at `SUP_HOME\Servers\UnwiredServer\logs\ClientTrace\`.

Debugging Replication Synchronization

Describes the replication synchronization process flow, and options available for diagnosing problems at each step of the process.

Process flow

This diagram shows a logical view of replication synchronization communications between the enterprise information system (EIS), Sybase Unwired Platform components, and device clients. Relay server is included as an optional component.

Figure 5: Replication synchronization process flow

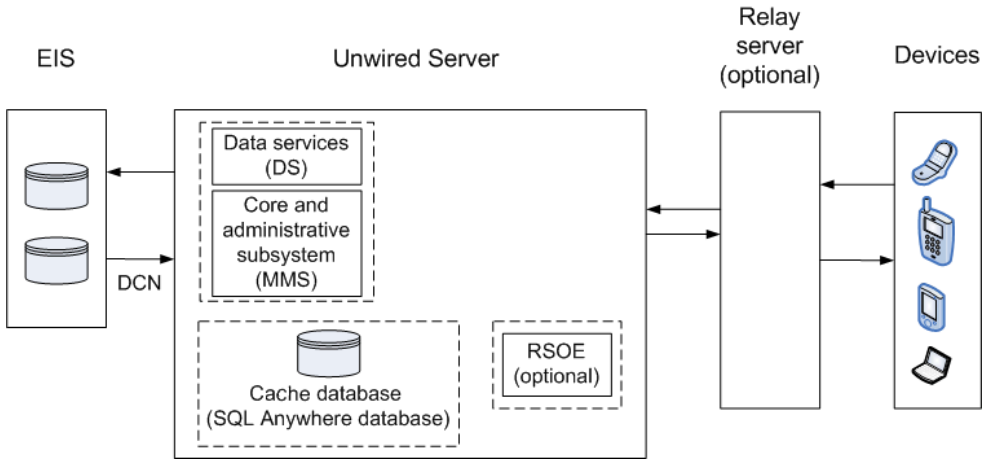
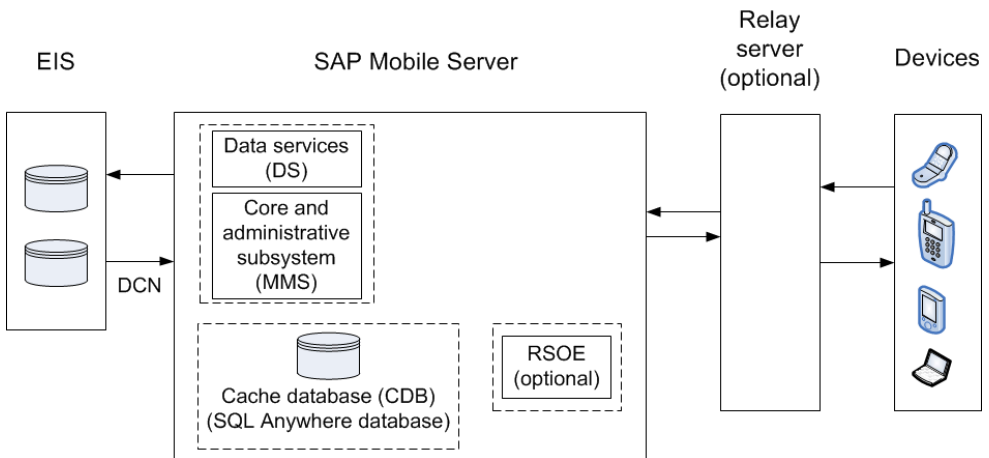


Figure 6: Replication synchronization process flow



Troubleshooting options

This section describes the options available for diagnosing replication synchronization processing flow problems.

Unwired Server – data services (DS):

- In Sybase Control Center, enable data services (DS) subsystem logging by setting its log level. The output is written in the server log.
- Review subsystem messages for information about getting data and sending transactions.

- For cache activities, you can enable domain logging on the package using Sybase Control Center, and then review data in cache sub-systems (Cache Refresh and DS Interface specifically) for the period domain logging is enabled. See *Domain Logs* and *Domain Log Categories* related documentation in *System Administration* and *Sybase Control Center for Sybase Unwired Platform* online help for complete details.

Unwired Server – core and administrative subsystem (MMS):

- If you like to troubleshoot at the system-level (which will record all domains and packages activities), you can enable MMS logging by setting the log level in Sybase Control Center. The output is written in the server log.
- Select the desired application package, and enable domain logging on that package. Review the captured log messages in Sybase Control Center.
- For quick troubleshooting to locate system errors, you can check the package-level client log and MBO/Operation history in the Sybase Control Center.
- View the MBO/Operation history in Sybase Control Center for errors during back-end interaction with the EIS.
- On the data-tier node, review the database error logs:
 - `errorlog.txt` – database system errors
 - `clusterdb_errorlog.txt` – cluster DB server log
 - `monitordb_errorlog.txt` – monitor and domainlog DB server log

Relay server – if used:

- Enable relay server tracing by setting verbosity to 1 or higher. You can do that by editing the `rs.config` file of your Relay Server and running the `rshost` utility. Check the *System Administration* topic on Relay Server Setup.
- Review the relay server log files. The logging for `rshost.exe` is directed to the `myrshost.log` file in the `C:\temp` directory.
- Check the IIS log files (`windows\logfiles\W3SVC1` and `HTTPERR`). The IIS server log path is configurable from `inetmgr` from **Default Web Site > Properties > Website tab > Enable Logging > Properties > Log file directory**. The recommended location is `C:\Inetpub\LogFiles`.

On Unwired Server, check the Sybase Unwired Platform outbound enabler log for replication synchronization entries: `SUP_HOME\Servers\UnwiredServer\Logs\NodeName.RSOEnum.log` (for example, `smpqa01SMPServer1a.RSOE1.log`).

Device clients:

- Use the platform-specific client logging API (**com.sybase.persistence.LogRecord** in Java, and **Sybase.Persistence.ILogRecord** in C#) to capture log messages and upload them (Client Log) to Unwired Server. You can view these messages in the Sybase Control Center user interface

Debugging the Cache

Describes the cache database cache process flow, and options available for diagnosing problems at each step of the process.

Process flow

This diagram shows a logical view of the cache process flow on Sybase Unwired Platform.

The cache refresh process is either scheduled (repeated at certain intervals) or on-demand when the client initiates a request. The cache can also be updated by incoming Data Change Notification (DCN) requests. A mobile business object (MBO) operation marked as `Invalidate the cache` operation forces cache contents to be refreshed.

Figure 7: Cache process flow

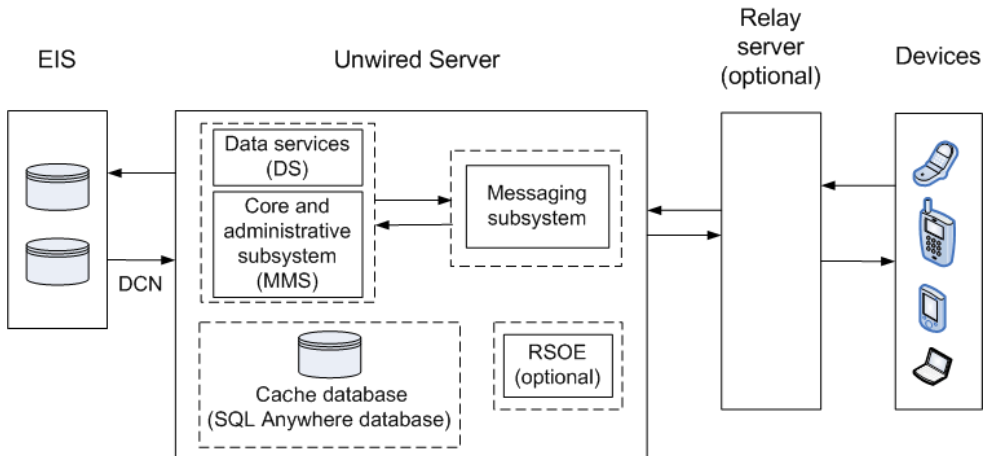
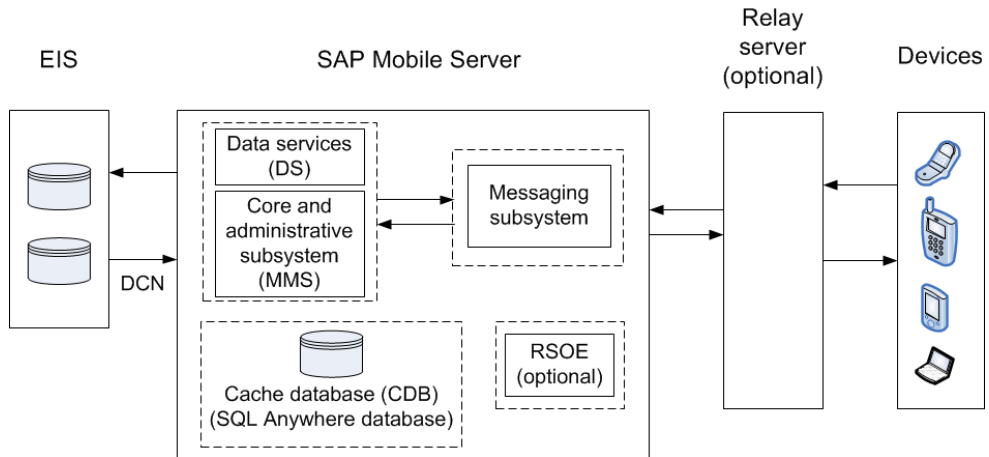


Figure 8: Cache process flow



Troubleshooting options

This section describes the options available for diagnosing cache database cache processing flow problems.

Unwired Server – data services (DS):

- In Sybase Control Center, enable data services (DS) subsystem logging by setting its log level. The output is written in the server log.
- Review subsystem messages for information about getting data and sending transactions.
- For cache activities, you can enable domain logging on the package using Sybase Control Center, and then review data in cache sub-systems (Cache Refresh and DS Interface specifically) for the period domain logging is enabled. See *Domain Logs* and *Domain Log Categories* related documentation in *System Administration* and *Sybase Control Center for Sybase Unwired Platform* online help for complete details.

In Sybase Control Center, check cache group properties such as schedule, and cache interval. Cache updates are propagated to the client based on a combination of settings for Synchronization group change detection interval, push related device settings (messaging synchronization), and notification threshold for subscriptions (replication synchronization). You can view all those properties in the Sybase Control Center .

You can also use monitoring to trace the cache refresh timestamp, notification generation timestamp, data import message generation (MBS), or sync request from the client for local database update. See *System Administration* for further details on cache.

Note: Monitoring is a useful tool, but keep in mind that it places a load on the system. When monitoring the cache database (CDB), cluster database (clusterDB), and DomainLog/Monitor database, set a maximum cache size, to limit automatic cache growth, using the SQL Anywhere option `-ch dbeng12/dbsrv12 server`. Otherwise, memory usage continues to grow, and may cause Unwired Server to fail.

Troubleshoot Basic Scenarios

Unwired Server – Core and administrative subsystem (MMS):

- If you like to troubleshoot at the system-level (which will record all domains and packages activities), you can enable MMS logging by setting the log level in Sybase Control Center. The output is written in the server log.
- Select the desired application package, and enable domain logging on that package. Review the captured log messages in Sybase Control Center.
- For quick troubleshooting to locate system errors, you can check the package-level client log and MBO/Operation history in the Sybase Control Center.
- View the MBO/Operation history in Sybase Control Center for errors during back-end interaction with the EIS.
- On the data-tier node, review the database error logs:
 - `errorlog.txt` – database system errors
 - `clusterdb_errorlog.txt` – cluster DB server log
 - `monitordb_errorlog.txt` – monitor and domainlog DB server log

Debugging Data Change Notification

Describes the data change notification process flow, and options available for diagnosing problems at each step of the process.

Process flow

This diagram shows a logical view of data change notification (DCN) between the enterprise information system (EIS) and Sybase Unwired Platform components.

DCNs notify Unwired Server when data used by an application changes on an EIS server. DCNs are used as a strategy to update very specific data automatically. See *Developer Guide: Unwired Server Runtime* and *System Administration* for development and administration information. The *Developer Guide: Unwired Server Runtime* provides DCN format and other details.

Figure 9: DCN process flow

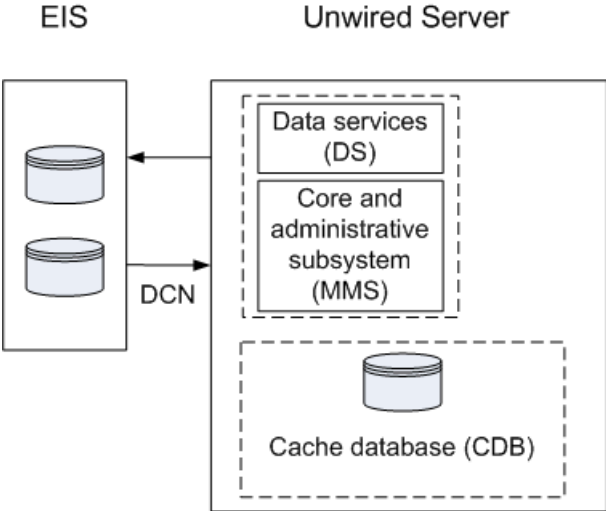
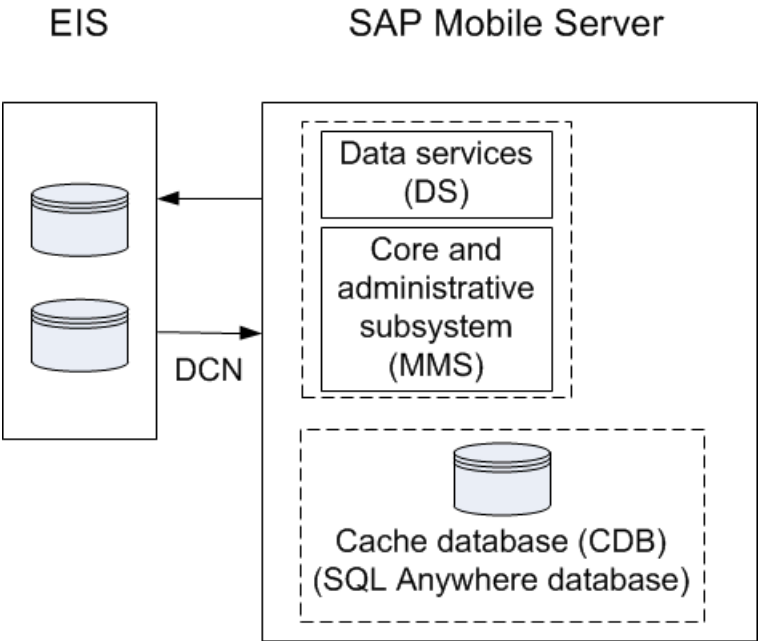


Figure 10: DCN process flow



DCN Error Reporting

Errors while processing data change notification (DCN) requests are reported with HTTP error codes. A single DCN request can contain multiple DCN messages and each message can have payload for a specific mobile business object (MBO).

If one of the DCN messages in a DCN request fail, then all the other successfully executed DCN messages are rolled back and a failure is reported as a DCN result with individual status messages for each DCN message. DCN messages that occur after the failed message are not processed, and failure is reported for them as well.

Success is reported only if all the DCN messages succeed.

Table 4. Error codes applicable to DCN

Error code	Error reason	Example of error cause
400	DCN usage error	A required parameter is missing from the DCN request.
401	Authentication error	The user name or password is incorrect.
403	Authorization error	The user does not have the SUPDCNUser assigned.
500	Internal server error	The DCN filter could not be loaded.

Table 5. DCN usage errors, HTTP error 400

Error	Error message
JSON parsing error	DCN JSON request <code><jsonDCNRequest></code> could not be parsed. Make sure it conforms to DCN JSON standard.
Required parameters not specified in DCN request like operation name, package name, user-name, password, dcn_request	<code><parameter></code> not supplied in the DCN request.
Wrong MBO name in DCN request	VirtualTableName is null. MBO name <code><mbo-Name></code> might be incorrect or missing capitalization in the DCN request.

Table 6. SSybase Unwired Platform internal errors, HTTP error code 500

Error code	Error reason
Wrong attributes in DCN request	Unable to set attribute names to update the cache. Check the cols section in the DCN request.

Error code	Error reason
DS error in modifying cache	As returned by the DS.
DCN filter classloading issues	Package <packageName> or Server wide loader for filter class <filterClassName> not found. Check if DCN request contains the fully qualified name of the filter class and/or it is deployed on the Unwired Server.
DCN filter processing errors	Error in postprocessing/preprocessing by filter class <classname>.
Error in setting domain	Error in setting the current domain
Package properties not found in the Sybase Unwired Platform repository	Package properties not found for package <packageName> in domain <domainName>.
Error in getting DS object to process DCN	dsAPIServiceNotFound.
Error dispatching cache event by DS	As returned by the DS.

Troubleshooting options

This section describes the options available for diagnosing DCN processing flow problems.

EIS

- Verify the EIS data source (third party application, database, Web service) is running.
- Verify the DCN code is operating correctly.

Note: If the DCN request is sent in HTTP, you may simply construct an HTTP request and send them through an Internet browser, instead of running from Java code and so forth, and then continue with other debugging steps.

Similar to cache, updates from DCN activity are propagated to devices based on a number of other settings. Review the change detection interval property associated with the synchronization group, the notification threshold of replication synchronization subscription, or the push-related settings for the device (message synchronization).

Unwired Server – data Services (DS):

- In Sybase Control Center, enable data services (DS) subsystem logging by setting its log level. The output is written in the server log.
- Review subsystem messages for information about getting data and sending transactions.
- For cache activities, you can enable domain logging on the package using Sybase Control Center, and then review data in cache sub-systems (Cache Refresh and DS Interface specifically) for the period domain logging is enabled. See *Domain Logs* and *Domain Log*

Categories related documentation in *System Administration* and *Sybase Control Center for Sybase Unwired Platform* online help for complete details.

Unwired Server – administrative Services (MMS):

- If you like to troubleshoot at the system-level (which will record all domain and package activities), you can enable MMS logging by setting the log level in Sybase Control Center. The output is written in the server log.
- Select the desired application package, and enable monitoring. The captured data appears in Sybase Control Center. Use package-level logging to review data in the domain log.
- On Unwired Server, review the `errorlog.txt` for any cache database system errors: `SUP_HOME\Servers\UnwiredServer\logs\errorlog.txt`.
- Enable domain logging on the package to log incoming DCN requests.
- Check the DCN request log in the Unwired Server HTTP Request Log (`SUP_HOME\Servers\UnwiredServer\logs\hostname-http.log`). The log shows a history of DCN requests, and performance data (including counts of total requests and processing time).

Debugging Android Device Development

Device client and Unwired Server troubleshooting tools for diagnosing Google Android development problems.

Client-side debugging

Problems on the device client side that may cause client application problems:

- Unwired Server connection failed.
- Data does not appear on the client device.
- Physical device problems, such as low battery or low memory.

To find out more information on the device client side:

- If you have implemented debugging in your generated or custom code (which Sybase recommends), turn on debugging, and review the debugging information. See *Developer Guide: Android Object API Applications* about using the `MBOLogger` class to add log levels to messages reported to the console.
- Check the log record on the device. Use the `<PkgName>DB.getLogRecords` (`com.sybase.persistence.Query`) or `Entity.getLogRecords()` methods. Use this method for logs corresponding to MBO classes.

This is the log format:

```
level,code,eisCode,message,component,entityKey,operation,requestId,timestamp
```

This is a log sample:

```
5,500,'','java.lang.SecurityException:Authorization failed:
Domain = default Package = end2end.rdb:1.0 mboName =
```

```
simpleCustomer action =
delete,'simpleCustomer','100001','delete','100014','2010-05-11
14:45:59.710'
```

- `level` – the log level currently set. Values include: 1 = TRACE, 2 = DEBUG, 3 = INFO, 4 = WARN, 5 = ERROR, 6 = FATAL, 7 = OFF.
- `code` – replication synchronization, Unwired Server administration codes, based on HTTP return codes:
 - 200 – success.
 - 401 – the client request included invalid credentials, or authentication failed for some other reason.
 - 404 – the client tried to access a nonexistent package or MBO.
 - 409 – backend EIS is deadlocked.
 - 412 – backend EIS threw a constraint exception.
 - 500 – failure.
 - 503 – backend EIS is not responding, or the connection is terminated.
 - 601 – the cache database (CDB) reported an error.
- `eisCode` – a value specific to the EIS backend, if any. Typically defaults to blank or 0 (zero); can be set via result checker.
- `message` – the message content.
- `component` – Mobile Business Object (MBO) name.
- `entityKey` – MBO surrogate key, used to identify and track MBO instances and data.
- `operation` – operation name.
- `requestId` – operation replay request ID or messaging-based synchronization message request ID.
- `timestamp` – message logged time, or operation execution time.
- For other issues, you can use tracing on the device side to trace Client Object API activity. Sybase Unwired Platform supports three levels of tracing:
 - 1 = Tracing is disabled
 - 3 = SQL Trace without payload
 - 5 = SQL Trace with payload (values displayed)

There are two ways to enable tracing on the client side:

- Via Sybase Control Center through the Applications folders:
 1. Click on the Applications folder in the Unwired Platform Cluster frame.
 2. In the **Applications** tab, select the application you wish to enable tracing for.
 3. Select the **Application Connections** tab and select the user you wish to enable tracing for.
 4. Click on the **Properties** button to open the Application Connection Properties window.
 5. Click on **Device Advanced** to display advanced connection properties.

6. Change the value of the **Device Level Trace** property to the appropriate level.
 7. Click **OK** to save your changes.
- Via code by calling the appropriate API. In the `SUPConnectionProfile` class, there are two APIs that can be used:
 - `- enableTrace(boolean enable)`
 - `- enableTrace(boolean enable, boolean with Payload)`

The application in the initialization code can invoke these APIs to enable/disable tracing:

```
ConnectionProfile *cp = SUP101DB.getConnectionProfile();

// To enable trace of client database operations (SQL
statements, etc.)
cp.enableTrace(true);

// To enable trace of client database operations with values
also displayed
cp.enableTrace(true, true);

// To disable trace of client database operations
cp.enableTrace(false);
```

Note: Once the trace level is set, the application must be restarted for the tracing to take effect. SQL tracing (level 3 and 5) can cause significant performance degradation, so it should be used with caution. SQL tracing should only be enabled via Sybase Control Center long enough for the device application to capture relevant tracing. When finished, you should shut down the application and relaunch it to run with tracing disabled.

Server-side debugging

Problems on the Unwired Server side that may cause device client problems:

- The domain or package does not exist.
- Authentication failed for the synchronizing user.
- The operation role check failed for the synchronizing user.
- Back-end authentication failed.
- An operation failed on the remote, replication database back end, for example, a table or foreign key does not exist. Detailed messages can be found in the Log Record.
- An operation failed on the Web service, REST, or SAP® back end. You can find detailed messages in the log record.

To find out more information on the Unwired Server side:

- Check the MMS server log files. See the *Sybase Control Center* documentation for more information.

Debugging BlackBerry Device Development

Device client and Unwired Server troubleshooting tools for diagnosing RIM® BlackBerry® development problems.

Client-side debugging

Problems on the device client side that may cause client application problems:

- Unwired Server connection failed.
- Data does not appear on the client device.
- Physical device problems, such as low battery or low memory.

To find out more information on the device client side:

- If you have implemented debugging in your generated or custom code (which Sybase recommends), turn on debugging, and review the debugging information. See *Developer Guide: BlackBerry Object API Applications* about using the MBOLogger class to add log levels to messages reported to the console.
- Check the log record on the device. Use the `<PkgName>DB.getLogRecords` (**com.sybase.persistence.Query**) or **Entity.getLogRecords()** methods. Use this method for logs corresponding to MBO classes.

This is the log format:

```
level,code,eisCode,message,component,entityKey,operation,requestId,timestamp
```

This is a log sample:

```
5,500,',','java.lang.SecurityException:Authorization failed:
Domain = default Package = end2end.rdb:1.0 mboName =
simpleCustomer action =
delete','simpleCustomer','100001','delete','100014','2010-05-11
14:45:59.710'
```

- `level` – the log level currently set. Values include: 1 = TRACE, 2 = DEBUG, 3 = INFO, 4 = WARN, 5 = ERROR, 6 = FATAL, 7 = OFF.
- `code` – replication synchronization, Unwired Server administration codes, based on HTTP return codes:
 - 200 – success.
 - 401 – the client request included invalid credentials, or authentication failed for some other reason.
 - 404 – the client tried to access a nonexistent package or MBO.
 - 409 – backend EIS is deadlocked.
 - 412 – backend EIS threw a constraint exception.
 - 500 – failure.

Troubleshoot Basic Scenarios

- 503 – backend EIS is not responding, or the connection is terminated.
 - 601 – the cache database (CDB) reported an error.
 - `eisCode` – a value specific to the EIS backend, if any. Typically defaults to blank or 0 (zero); can be set via result checker.
 - `message` – the message content.
 - `component` – Mobile Business Object (MBO) name.
 - `entityKey` – MBO surrogate key, used to identify and track MBO instances and data.
 - `operation` – operation name.
 - `requestId` – operation replay request ID or messaging-based synchronization message request ID.
 - `timestamp` – message logged time, or operation execution time.
 - Check the Storm event log:
 1. On the Home screen, press Hold.
 2. Click the upper-left corner and upper-right corner twice.
 3. Review the event log.
 - Check the BlackBerry event log:
 1. On the device, press ALT+Iglg; or, for touch-screen devices, hold the ESC key, tap (no click) top-left, top-right, top-left, then top-right.
 2. Review the event log, and see the RIM BlackBerry documentation for information about debugging and optimizing. http://na.blackberry.com/eng/developers/resources/A50_How_to_Debug_and_Optimize_V2.pdf
 - For other issues, you can use tracing on the device side to trace Client Object API activity. Sybase Unwired Platform supports three levels of tracing:
 - 1 = Tracing is disabled
 - 3 = SQL Trace without payload
 - 5 = SQL Trace with payload (values displayed)
- There are two ways to enable tracing on the client side:
- Via Sybase Control Center through the Applications folders:
 1. Click on the Applications folder in the Unwired Platform Cluster frame.
 2. In the **Applications** tab, select the application you wish to enable tracing for.
 3. Select the **Application Connections** tab and select the user you wish to enable tracing for.
 4. Click on the **Properties** button to open the Application Connection Properties window.
 5. Click on **Device Advanced** to display advanced connection properties.
 6. Change the value of the **Device Level Trace** property to the appropriate level.
 7. Click **OK** to save your changes.
 - Via code by calling the appropriate API. In the `SUPConnectionProfile` class, there are two APIs that can be used:

- `- enableTrace(boolean enable)`
- `- enableTrace(boolean enable, boolean withPayload)`

The application in the initialization code can invoke these APIs to enable/disable tracing:

```
ConnectionProfile *cp = SUP101DB.getConnectionProfile();

// To enable trace of client database operations (SQL
statements, etc.)
cp enableTrace(true);

// To enable trace of client database operations with values
also displayed
cp enableTrace(true, true);

// To disable trace of client database operations
cp enableTrace(false);
```

Note: Once the trace level is set, the application must be restarted for the tracing to take effect. SQL tracing (level 3 and 5) can cause significant performance degradation, so it should be used with caution. SQL tracing should only be enabled via Sybase Control Center long enough for the device application to capture relevant tracing. When finished, you should shut down the application and relaunch it to run with tracing disabled.

Server-side debugging

Problems on the Unwired Server side that may cause device client problems:

- The domain or package does not exist.
- Authentication failed for the synchronizing user.
- The operation role check failed for the synchronizing user.
- Back-end authentication failed.
- An operation failed on the remote, replication database back end, for example, a table or foreign key does not exist. Detailed messages can be found in the Log Record.
- An operation failed on the Web service, REST, or SAP® back end. You can find detailed messages in the log record.

To find out more information on the Unwired Server side:

- Check the MMS server log files. See the *Sybase Control Center* documentation for more information.

Debugging iOS Device Development

Device client and Unwired Server troubleshooting tools for diagnosing Apple iOS development problems.

Client-Side Debugging

Problems on the device client side that may cause client application problems:

Troubleshoot Basic Scenarios

- Unwired Server connection failed.
- Data does not appear on the client device.
- Physical device problems, such as low battery or low memory.

To find out more information on the device client side:

- If you have implemented debugging in your generated or custom code (which Sybase recommends), turn on debugging, and review the debugging information. See *Developer Guide: iOS Object API Applications* about using the `MBOLogger` class to add log levels to messages reported to the console.
- Check the log record on the device. Use the individual entity class `getLogRecords` method, or assemble a query using the `SUPQuery` class and call the database class `getLogRecords` method.

This is a sample log format (depending on your query):

```
<header> [level]
        code
        component
        entityKey
        level
        message
        operation
        requestId
        timestamp
```

This log format generates output similar to:

```
2010-06-22 10:54:28.927 SAPSimpleApps2[91202:207] [INFO]
[SAPTest.m:295] entity has a log record:
        code = 200,
        component = SalesOrder,
        entityKey = 500151,
        level = 3,
        message = Successful,
        operation = changeFromData,
        requestId = 33416006,
        timestamp = 2010-06-22 03:54:22 -0700
```

- `code` – log record code numbers sent by Unwired Server, based on HTTP return codes.
 - Replication synchronization codes:
 - 200 – success.
 - 500 – failure.
 - Message synchronization codes:
 - 200 – success.
 - 401 – the client request included invalid credentials, or authentication failed for some other reason.

- 403 – the client request included valid credentials, but the user does not have permission to access the requested resource (package, mobile business object —MBO, or operation).
- 404 – the client tried to access a nonexistent package or MBO.
- 405 – there is no valid license to check out for the client.
- 409 – back-end EIS is deadlocked.
- 412 – back-end EIS threw a constraint exception.
- 500 – an unexpected (unspecified) server failure.
- 503 – back-end EIS is not responding or the connection is terminated.
- `eisCode` – a value specific to the EIS backend, if any. Typically defaults to blank or 0 (zero); can be set via result checker.
- `entityKey` – if this log record refers to a particular row in an MBO table, this field contains a string representation of the primary key for the row.
- `component` – if this log record refers to a particular row in an MBO table, this field contains the name of the MBO.
- `level` – the log level currently set. Values include: 1 = Trace, 2 = Debug, 3 = Info, 4 = Warn, 5 = Error, and 6 = Fatal.
- `message` – the reason for the log record, for example, an error from the server when attempting to process a client replay request for a create, update, and delete operation.
- `messageID` – unique log record ID.
- `operation` – for log records written by Unwired Server, the operation (such as create, update, and delete) that caused this log record to be written.
- `requestId` – for log records written by Unwired Server, the ID of the replay request that caused this log record to be written.
- `timestamp` – the date and time the log record was created.
- If you have implemented `onConnectionStatusChange` for message synchronization in `CallbackHandler`, the connection status between Unwired Server and the device is reported on the device. See the *Developer Guide: iOS Object API Applications* for `CallbackHandler` information. The status reported on the device includes:
 - 1 – device is connected.
 - 2 – device is not connected.
 - 3 – device is not connected because of flight mode.
 - 4 – device is not connected because no network coverage.
 - 5 – device is not connected and waiting to retry a connection.
 - 6 – device is not connected because roaming was set to false and device is roaming.
 - 7 – device is not connected because of low space.
- For other issues, you can use tracing on the device side to trace Client Object API activity. Sybase Unwired Platform supports three levels of tracing:
 - 1 = Tracing is disabled

Troubleshoot Basic Scenarios

- 3 = SQL Trace without payload
- 5 = SQL Trace with payload (values displayed)

There are two ways to enable tracing on the client side:

- Via Sybase Control Center through the Applications folders:
 1. Click on the Applications folder in the Unwired Platform Cluster frame.
 2. In the **Applications** tab, select the application you wish to enable tracing for.
 3. Select the **Application Connections** tab and select the user you wish to enable tracing for.
 4. Click on the **Properties** button to open the Application Connection Properties window.
 5. Click on **Device Advanced** to display advanced connection properties.
 6. Change the value of the **Device Level Trace** property to the appropriate level.
 7. Click **OK** to save your changes.
- Via code by calling the appropriate API. In the SUPConnectionProfile class, there are two APIs that can be used:
 - (void) enableTrace:(BOOL)enable
 - (void) enableTrace:(BOOL)enable withPayload:(BOOL)withPayload

The application in the initialization code can invoke these APIs to enable/disable tracing:

```
SUPConnectionProfile *cp = [SUP101SUP101DB
getConnectionProfile];

// To enable trace of client database operations (SQL
statements, etc.)
[cp enableTrace:YES];

// To enable trace of client database operations with values
also displayed
[cp enableTrace:YES withPayload:YES];

// To disable trace of client database operations
[cp enableTrace:NO];

// To enable trace of message headers sent to the server and
received from the server
// (this replaces the MBODebugLogger and MBODebugSettings used
in earlier versions of SUP)
[cp.syncProfile enableTrace:YES];

// To enable trace of both message headers and content,
including credentials
[cp.syncProfile enableTrace:YES withPayload:YES];

// To disable messaging trace
[cp.syncProfile enableTrace:NO];
```

Note: Once the trace level is set, the application must be restarted for the tracing to take effect. SQL tracing (level 3 and 5) can cause significant performance degradation, so it

should be used with caution. SQL tracing should only be enabled via Sybase Control Center long enough for the device application to capture relevant tracing. When finished, you should shut down the application and relaunch it to run with tracing disabled.

Server-Side Debugging

Problems on the Unwired Server side that may cause device client problems:

- The domain or package does not exist.
- Authentication failed for the synchronizing user.
- The operation role check failed for the synchronizing user.
- Back-end authentication failed.
- An operation failed on the remote, replication database back end, for example, a table or foreign key does not exist.
- An operation failed on the Web Service, REST, or SAP back end. You can find detailed messages in the log record.

To find out more information on the Unwired Server side:

- Check the Unwired Server log files.
- For message-based synchronization mode, check the SUPBridge and JMSBridge logs.

Debugging Windows and Windows Mobile Object API Development

Device client and Unwired Server troubleshooting tools for diagnosing Microsoft Windows and Windows Mobile development problems.

Client-Side Debugging

Problems on the device client side that may cause client application problems:

- Unwired Server connection failed.
- Data does not appear on the client device.
- Physical device problems, such as low battery or low memory.

To find out more information on the device client side:

- If you have implemented debugging in your generated or custom code (which Sybase recommends), turn on debugging and review the debugging information. See *Developer Guide: Windows and Windows Mobile Object API Applications* about using the MBOLogger class to add log levels to messages reported to the console.
- Check the log record on the device. Use the `DatabaseClass.GetLogRecord(Sybase.Persistence.Query)` or `Entity.GetLogRecords` methods.

This is the log format

Troubleshoot Basic Scenarios

```
level,code,eisCode,message,component,entityKey,operation,requestId,timestamp
```

This log format generates output similar to:

```
level code eisCode message component entityKey operation requestId
timestamp
5,500,'','java.lang.SecurityException:Authorization failed:
Domain = default Package = end2end.rdb:1.0 mboName =
simpleCustomer action =
delete','simpleCustomer','100001','delete','100014','2010-05-11
14:45:59.710'
```

- `level` – the log level currently set. Values include: 1 = TRACE, 2 = DEBUG, 3 = INFO, 4 = WARN, 5 = ERROR, 6 = FATAL, 7 = OFF.
 - `code` – Unwired Server administration codes, based on HTTP return codes.
 - Replication synchronization codes:
 - 200 – success.
 - 401 – the client request included invalid credentials, or authentication failed for some other reason.
 - 404 – the client tried to access a nonexistent package or MBO.
 - 409 – backend EIS is deadlocked.
 - 412 – backend EIS threw a constraint exception.
 - 500 – failure.
 - 503 – backend EIS is not responding, or the connection is terminated.
 - 601 – the cache database (CDB) reported an error.
 - `eisCode` – a value specific to the EIS backend, if any. Typically defaults to blank or 0 (zero); can be set via result checker.
 - `message` – the message content.
 - `component` – MBO name.
 - `entityKey` – MBO surrogate key, used to identify and track MBO instances and data.
 - `operation` – operation name.
 - `requestId` – operation replay request ID or messaging-based synchronization message request ID.
 - `timestamp` – message logged time, or operation execution time.
- For other issues, you can use tracing on the device side to trace Client Object API activity. Sybase Unwired Platform supports three levels of tracing:
- 1 = Tracing is disabled
 - 3 = SQL Trace without payload
 - 5 = SQL Trace with payload (values displayed)

There are two ways to enable tracing on the client side:

- Via Sybase Control Center through the Applications folders:

1. Click on the Applications folder in the Sybase Unwired Platform Cluster frame.
 2. In the **Applications** tab, select the application you wish to enable tracing for.
 3. Select the **Application Connections** tab and select the user you wish to enable tracing for.
 4. Click on the **Properties** button to open the Application Connection Properties window.
 5. Click on **Device Advanced** to display advanced connection properties.
 6. Change the value of the **Device Level Trace** property to the appropriate level.
 7. Click **OK** to save your changes.
- Via code by calling the appropriate API. In the SUPConnectionProfile class, there are two APIs that can be used:
 - ```
- EnableTrace(bool enable)
- EnableTrace(bool enable, bool withPayload)
```

The application in the initialization code can invoke these two APIs to enable/disable tracing:

```
ConnectionProfile cp = SampleDB.GetConnectionProfile();

// To enable trace of client database operations (SQL
statements, etc.)
cp.EnableTrace(true);

// To enable trace of client database operations with values
also displayed
cp.EnableTrace(true,true);

// To disable trace of client database operations
cp.EnableTrace(false);
```

**Note:** Once the trace level is set, the application must be restarted for the tracing to take effect. SQL tracing (level 3 and 5) can cause significant performance degradation, so it should be used with caution. SQL tracing should only be enabled via Sybase Control Center long enough for the device application to capture relevant tracing. When finished, you should shut down the application and relaunch it to run with tracing disabled.

### *Server-Side Debugging*

Problems on the Unwired Server side that may cause device client problems:

- The domain or package does not exist. If you create a new domain, whose default status is disabled, it is unavailable until enabled.
- Authentication failed for the synchronizing user.
- The operation role check failed for the synchronizing user.
- Back-end authentication failed.
- An operation failed on the remote, replication database back end, for example, a table or foreign key does not exist.

## Troubleshoot Basic Scenarios

- An operation failed on the Web Service, REST, or SAP® back end.

To find out more information on the Unwired Server side:

- Check the Unwired Server log files.
- You can also obtain DEBUG information for a specific device:
  - View information through the Sybase Control Center:
    1. Set the DEBUG level to a higher value for a specified device through Sybase Control Center:
      - a. On Sybase Control Center, select **Application Connections**, then select **Properties... > Device Advanced**.
      - b. Set the Debug Trace Level value.
    2. Set the TRACE file size to be more than 50KB.
    3. View the trace file through Sybase Control Center.
  - Check the `SUP_HOME\Servers\UnwiredServer\logs\ClientTrace` directory to see the mobile device client log files for information about a specific device.

---

**Note:** It is important to return to INFO mode as soon as possible, since DEBUG mode can affect system performance.

---

## Debugging DOE-C End-to-End Process Flow

---

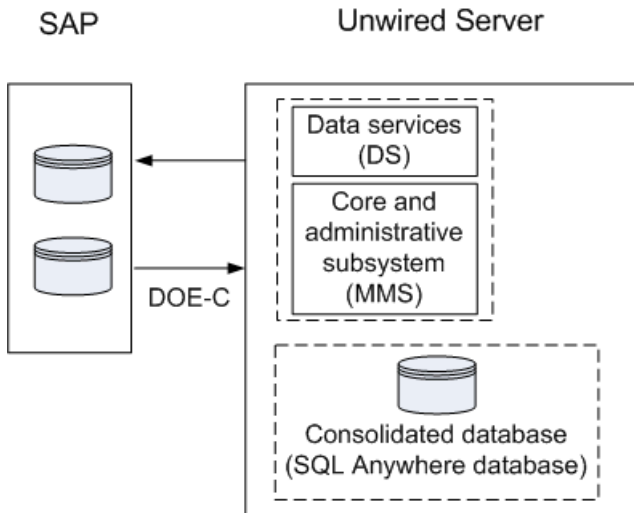
Troubleshooting tools for debugging end-to-end processing problems using a DOE connection and SAP data source.

### *High-level Process Flow*

This section describes high-level process flow when DOE-C is configured with Sybase Unwired Platform.



**Figure 11: DOE-C Process Flow**



*Troubleshooting Options*

This section describes the options available for diagnosing process flow problems.

*SAP-side Debugging*

- Verify the SAP data source is running.
- Verify the SAP data source privileges have been granted properly.
- Verify credentials are correct.
- Check for error messages.

*DOE Connector (DOE-C) Debugging*

- Verify the DOE-C connection is operational.
- Verify credentials are correct.
- Check for error messages.

*Sybase Unwired Platform-side Debugging*

- Verify Unwired Server is up and running.
- Verify the DOE-C connection is configured properly.

## Debugging JCo End-to-End Process Flow

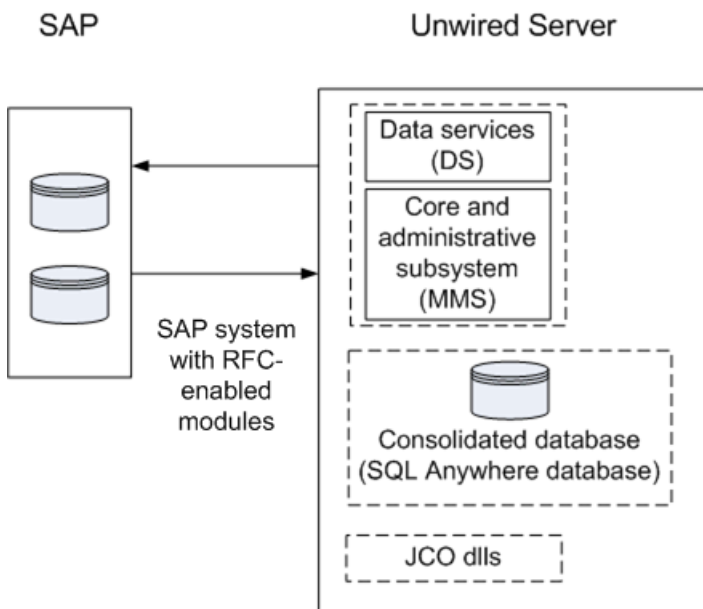
---

Troubleshooting options for debugging end-to-end processing problems using a JCo connection to an SAP data source and Web service to expose SAP BAPIs.

### *High-level Process Flow*

This section describes high-level process flow when using SAP as a data source.

**Figure 12: SAP JCo Connection Process Flow**



### *Troubleshooting Options*

This section describes the options available for diagnosing process flow problems.

#### *SAP-side Debugging*

- Verify the SAP data source is running.
- Verify the SAP data source privileges have been granted properly.
- Ensure the SAP BAPIs are exposed properly as Web services.
- Verify credentials are correct.
- Check for error messages.

Search *Troubleshooting* for the topic *Verifying SAP Prerequisites*.

### *SAP JCo Connection Debugging*

- Verify the JCo connection is configured properly and is operational.
- Check for error messages.

### *Sybase Unwired Platform-side Debugging*

- Verify Unwired Server is up and running.
- Turn on JCo tracing in the Unwired Server system properties:
  1. In Sybase Control Center for Sybase Unwired Platform, select **Performance Configuration > User Options** for your Unwired Server. You can configure JVM parameters in the form `-Dxxx=xxx`. Add the following JVM parameters:

```
-Djco.trace_level=[0-10]
-Djco.trace_path=[defined_path]
```

The `jco.trace_level` determines the level of detail of tracing you get from the JCo libraries. The `jco.trace_path` determines a directory where trace files get stored. If your SAP-based MBOs fail, turning on tracing can help determine the cause.

- Verify the JCo connection is configured properly.

For Unwired Server JCo connections, see the topic *Preparing Unwired Server to Connect to SAP using Java Connectors* in *System Administration*.



# Troubleshoot the System

Provides troubleshooting information for the Sybase Unwired Platform system.

## Troubleshoot the Postinstallation Configuration

Provides troubleshooting information for postinstallation configuration problems in Sybase Unwired Platform.

### Troubleshooting Host Name Problems

**Problem:** After installation, you determine that you need to change the host name (machine name); for example, the host name is too long, or includes reserved characters. Changing the host name is rarely done in a production system.

**Solution:** Change the Unwired Server host name. For a node in a cluster, you must change the name for each node in the cluster. Additional steps are required if you are changing the host name after an upgrade from a previous version of Sybase Unwired Platform.

After changing the host name, you may encounter temporary connection problems to Unwired Server. This is because the DNS servers may not resolve the IP difference until the connections are refreshed.

#### Changing Unwired Server Host Name

Change the Unwired Server host name. If you are changing the host name for a node in a cluster, the name must be changed for all nodes in the cluster. Additional steps are required if you are making a host name change after an upgrade.

1. Stop the Unwired Server service.
2. Update all the `socketlistener` property files related to the node with the changed host name. Change the value of `host` to the new host name. The property files are located at `SUP_HOME\Servers\UnwiredServer\Repository\Instance\com\sybase\djc\server\SocketListener\<mlservername>_<protocol>.properties`. Open each file, and change the old host value name to the new host name.
3. Update the property value in the cluster database:
  - a) Using `dbisqlc`, connect to the cluster database.
  - b) Update the `sup.host` property value with the new node value. For example:
 

```
update MEMBER_PROP set value1='<new_hostname>' where name='sup.host' and value1='<old_hostname>'
```

4. Update the property value of `sup.host` with the new host name in `sup.properties`. It is located at: `Repository\Instance\com\sybase\sup\server\SUPServer\sup.properties`.
5. To enable Sybase Control Center to work:
  - a) Restart the Sybase Control Center *X.X* service.
  - b) Modify the URL in the shortcut for the Sybase Control Center. Use the new hostname instead of the old one in the launched URL
6. For a cluster environment, repeat step 2 for each node.
7. Restart the Unwired Server service.

### **Listeners or Services Could Not be Started Errors**

**Problem:** After installation, Listener could not be started, or Service could not be started messages appear.

**Explanation:** This may indicate your Sybase Unwired Platform configuration includes duplicate port numbers. This can happen for a variety of reasons; for example, if an application or service is offline during the installation process, the installer does not detect the reserved port number.

**Solution:** Change the Sybase Unwired Platform component port numbers as needed post-installation, using information in *Port Number Reference*.

#### **Sybase Control Center Could Not be Started**

**Problem:** After installation, Sybase Control Center could not be started. The agent bootstrap thread logs error messages in `agent.log` such as the ones below. In case of a port conflict on the RMI agent port (default 9999) or the Repository DB port (default 3638), the Sybase Control Center service startup fails.

```
RMI Agent:
[Agent Bootstrap Thread] - Failed to start rmi registry.
Messaging:
[Agent Bootstrap Thread] - Messaging provider failed to fully start.
Web Container:
[Agent Bootstrap Thread] - failed...java.net.BindException: Address
already in use: JVM_Bind
Repository DB:
[Agent Bootstrap Thread] - Failed to connect to dataserver engine.
```

**Explanation:** This error can occur for a variety of reasons; for example, if an application or service is offline during the installation process, the installer does not detect the port number is in use.

**Solution:** Change the Sybase Control Center component port number as needed post-installation, using information in *Port Number Reference*.

**Port Number Reference**

Change Sybase Unwired Platform component port numbers after installation, if necessary.

Proceed with caution when changing port numbers because the change might impact other configuration files that point to that port. You need to be aware of the default Sybase Control Center port numbers so you do not accidentally use these ports when you change Sybase Unwired Platform ports. You can change some Sybase Control Center default ports, but, in some cases, you should not.

---

**Note:** To make Unwired Server port number changes, temporarily stop the other service consuming those ports. Use Sybase Control Center to make the changes, then restart Unwired Server.

---

| Port                           | Description                                                                                                           | Default Port                                  | Instructions for Changing                                                                                                                                                                                                                                                                                              |
|--------------------------------|-----------------------------------------------------------------------------------------------------------------------|-----------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Data tier (CDB) server         | Port number for the data tier that manages transactions between the enterprise information system and mobile devices. | 5200                                          | Do not change the CDB port.                                                                                                                                                                                                                                                                                            |
| Management ports               | IIOP port number on which the Unwired Server listens for Sybase Control Center administration requests.               | 2000<br>2001 for secure management ( default) | Default is recommended. No change is required.                                                                                                                                                                                                                                                                         |
| Data change notification (DCN) | Port number on which Unwired Server listens for DCN events.                                                           | 8000 for HTTP<br>8001 for HTTPS               | Configure in Sybase Control Center by expanding the Servers > <ServerName> folder and selecting Server Configuration. In the General tab, select the Communication Ports subtab and enter a new DCN port or secure DCN port, as required.<br><br>See <i>Security Profiles</i> in in Sybase Control Center online help. |

| Port                       | Description                                                                                                                                                                                                                   | Default Port                                          | Instructions for Changing                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|----------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Synchronization            | <p>Port numbers on which Unwired Server synchronizes data between the enterprise information system and mobile devices.</p> <p>Messaging port uses a proprietary encryption method, so communication is always encrypted.</p> | <p>2480 for replication</p> <p>5001 for messaging</p> | <p>Configure in Sybase Control Center by expanding the Servers &gt; &lt;ServerName&gt; folder and selecting Server Configuration. In the Replication or Messaging tab, select the Synchronization Listener sub-tab and enter a new synchronization port, as required.</p> <hr/> <p><b>Note:</b> If there is a conflict for port 2480 or 2481, Unwired Server will not start, and you cannot use Sybase Control Center to modify them. To correct the problem, you must temporarily stop the service that uses the conflicting port, then start Unwired Server.</p> <hr/> <p>For replication payloads, see <i>Configuring Replication Properties</i> in Sybase Control Center online help.</p> <p>For messaging payloads, see <i>Configuring Messaging Properties</i> in in Sybase Control Center online help.</p> |
| Advantage Database Server® | Port number for the messaging database.                                                                                                                                                                                       | 6262                                                  | Sybase recommends that you do not change these ports.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |



| Port                                   | Description                                                                | Default Port                            | Instructions for Changing                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|----------------------------------------|----------------------------------------------------------------------------|-----------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Messaging server administration</p> | <p>Port number for the messaging service for Sybase messaging clients.</p> | <p>5100 for administration services</p> | <p>Cannot be changed in Sybase Control Center.</p> <p>Use the &lt;&lt;UnwiredPlatform_InstallDir&gt;&gt;\Servers\Messaging Server\Bin\AdminWebServicesTool.exe command line tool to change the messaging service Web service port. This tool has built in online help describing how to use the tool. From the command prompt run:</p> <pre>&lt;UnwiredPlatform_InstallDir&gt;\Servers\Messaging Server\Bin&gt;AdminWebServicesTool.exe set=&lt;port&gt; restart</pre> |

| Port                  | Description                                                                        | Default Port                                                                                                                                                                  | Instructions for Changing                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------------------|------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Sybase Control Center | Additional default port numbers of which to be aware, when modifying port numbers. | 9999 for default RMI agent port<br>2100 for default JMS messaging service port<br>3638 for default SCC repository database port<br>8282, 8283 for default Web container ports | <ul style="list-style-type: none"> <li>• 9999 – default RMI agent port. The port is set in:<br/> <code>&lt;&lt;UnwiredPlatform_InstallDir&gt;&gt;\SCC-XX\services\RMI\service-config.xml</code> </li> <li>• 2100 – default JMS messaging service port. The port is set in:<br/> <code>&lt;&lt;UnwiredPlatform_InstallDir&gt;&gt;\SCC-XX\services\Messaging\service-config.xml</code> </li> <li>• 3638 – default SCC repository database port. The default port is set in:<br/> <code>&lt;&lt;UnwiredPlatform_InstallDir&gt;&gt;\SCC-XX\services\ScsSADatasever\service-config.xml</code> </li> <li>• 8282, 8283 – default Web container ports. The default ports are set in:<br/> <code>&lt;&lt;UnwiredPlatform_InstallDir&gt;&gt;\SCC-XX\services\EmbeddedWebContainer\service-config.xml</code> </li> </ul> <p>Before you make any changes to these files, stop Sybase Control Center X.X service. Start the service after you complete the changes. If any of the subsystems fail to start, check the SCC agent.log for error messages.</p> |

| Port                      | Description                                                           | Default Port                                   | Instructions for Changing                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|---------------------------|-----------------------------------------------------------------------|------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Relay server              | Port numbers on which Relay Server listens for requests.              | 80 for the HTTP port<br>443 for the HTTPS port | Change the value for the cluster in Sybase Control Center for Unwired Platform. You can then generate the file and transfer it to the corresponding Unwired Server host.<br><br><i>See <b>Sybase Control Center &gt; Configure &gt; Configure Unwired Platform &gt; Clusters &gt; Relay Server.</b></i>                                                                                                                                                                                                                       |
| Unwired Platform reserved | Port numbers reserved for internal use by Unwired Platform components | 4343<br>5500<br>27000<br>8002<br>2638          | Do not use these special port numbers for any purpose. These ports differ from Windows reserved ports (1-1023).<br><br><b>Note:</b> Even if the installer does not detect a conflict at install time, Windows may later use ports in the 1024-64K range for other purposes. Read Microsoft documentation to determine how to reserve Unwired Platform ports. Otherwise, you may experience intermittent problems when starting up platform services due to Windows using these ports for some other purpose at the same time. |

**Viewing Assigned Ports**

Use Windows Task Manager and the Microsoft **netstat** utility to determine the process that is occupying a port in a Windows installation.

Check whether the ports are being used by other programs in a Windows environment:

1. Open Windows Task Manager, and select the **Processes** tab.
2. Click **View > Select Columns > .** From the list, select PID (Process Identifier), and click **OK**.
3. Access the command prompt (cmd), and type the following command:**netstat -aon**. The resulting list shows the PID, the IP addresses, and the ports.
4. Compare the PID in this list against the PID in the Task Manager list to identify the process which is occupying the port.

## **Sybase Unwired Server Service Fails to Restart After Credential Change**

**Problem:** The Sybase Unwired Server Service fails to restart after the person who installed Sybase Unwired Platform changes their login credentials.

**Explanation:** The Sybase Unwired Server service is installed using the login credentials of the user account of the person who runs the installer. After resetting their password as may be required by the security policy of their domain, the Sybase Unwired Server service fails to restart until the user updates the password for the login credentials configured for the service.

This behavior does not produce any Sybase Unwired Platform log entries to indicate a problem (there will not be any logs generated by the failed restart), however the Windows event viewer provides message similar to the following: "The SybaseUnwiredServer service was unable to log on as sybase\username with the currently configured password due to the following error: Logon failure: unknown user name or bad password."

**Solution:** Update the password for the login credentials configured for the service.

## **Handling Intrusion Detection/Prevention Software**

A personal firewall, or intrusion detection/prevention software (IPS or IDPS), can cause Unwired Platform components to malfunction or not function at all. Unwired Platform uses regular IP communication between components on the primary network interface of a computer, even when all components are installed on the same host.

If the local network interface is secured by intrusion detection/prevention software (IPS or IDPS, for example, McAfee Host Intrusion Prevention software or equivalent), you must configure the security software to allow all network communication between Unwired Platform components.

For a single-node installation of all of the Sybase Unwired Platform components, try one of these options to work around the limitations imposed by the host intrusion prevention software and policy settings, without violating any security policy, until the settings of your security software are adjusted to the needs of Unwired Platform.

Choose an option:

- Removing the host machine from the network – this option ensures that all interconnections between Sybase Unwired Platform components are treated as local traffic and is not be flagged as incoming connections from external sources, thereby causing connection failures due to security policy setting. This option is suitable when you use your laptop in a network other than your corporate network, and want to demonstrate a mobile solution using a simulator or emulator with all components running on the same machine. To use this option:
  1. Stop the Sybase Unwired Platform services in the correct order. See *Starting and Stopping Unwired Platform Server Services* in *System Administration*.

2. Disconnect the host from all networks.
  3. Restart Sybase Unwired Platform services in the correct order.
  4. Change the Sybase Control Center URL link to use "localhost" or *<yourhostname>* as the host name, instead of the original fully qualified host name of the machine that included the domain name (for example: `https://localhost:8283/scc`, or `https://yourhostname:8283/scc`). Accept any security warnings to connect to Sybase Control Center.
- Connecting the host to the corporate network – this option ensures that all interconnections among Sybase Unwired Platform components are internal to your corporate network and validated against the corporate network security policy. The option of connecting to corporate network through VPN is especially suitable when you use your laptop in a network other than your corporate network, and want to demonstrate a mobile solution using your physical devices, and need outgoing connections to a backend Enterprise Information System (EIS) or Relay Server (Sybase Hosted Relay Server or otherwise).
    1. Stop the Sybase Unwired Platform services in the correct order. See the *Starting and Stopping Unwired Platform Server Services* topic in the *System Administration*.
    2. Reconnect the host to your corporate network directly or through corporate VPN, to ensure that the corporate network security policy applies.
    3. Restart Sybase Unwired Platform services in the correct order.
    4. Change the Sybase Control Center URL link to use "localhost" or *<yourhostname>* as the host name, instead of the original fully qualified host name of the machine that included the domain name (for example: `https://localhost:8283/scc`, or `https://yourhostname:8283/scc`). Accept any security warnings to connect to Sybase Control Center.
  - Configuring the firewall software to allow connections to the ports the Unwired Platform uses. For a list of ports, see *Unwired Platform Ports* in *System Administration*.

Always check for the latest available patches and updates for your Unwired Server version on <http://downloads.sybase.com/swd/base.do?client=support>.

## **Cannot Access Device Tab and Web Service Error**

**Problem:** After installation, you cannot access the **Applications > Application Connections** tab in Sybase Control Center, and the Unwired Server log shows an ADMIN\_WEBSERVICE\_INVOCATION\_ERROR.

**Explanation:** This may indicate you have a port number conflict for the messaging administration Web services server listener port.

**Solution:** Change the port number.

1. Navigate to: `<UnwiredPlatform_InstallDir>\UnwiredPlatform\Servers\MessagingServer\Bin`
2. At the command line, enter:
 

```
AdminWebServicesTool.exe set=<port>
```

## Troubleshoot the System

Where `<port>` is the new port number. Alternatively, enter this to change the port number and restart the listener using the new number.

```
AdminWebServicesTool.exe set=<port> restart
```

### Cannot Access sampledb

**Problem:** After installation, you cannot access the sample database (sampledb).

**Explanation 1:** If you installed Unwired Platform with an Enterprise Server (production) license, this is expected behavior. You must start `sampledb` manually.

**Solution:** Use the Start and Stop `sampledb` command line utility (**samledb.bat**) to start the sample database. See *System Administration for Sybase Unwired Platform > System Reference > Command Line Utilities > Unwired Server Runtime Utilities > Start and Stop sampledb Server (sampledb) Utility*.

**Explanation 2:** If you installed Unwired Platform with an Enterprise or Personal Development license, make sure the `sampledb` windows services is installed properly and started. See *System Administration for Sybase Unwired Platform > System Reference > Command Line Utilities > Unwired Server Runtime Utilities > Start and Stop sampledb Server (sampledb) Utility*.

### System Environment Path is Too Long

**Problem:** When upgrading Sybase Unwired Platform, you cannot start Unwired Server using the `start-unwired-server.bat` batch file provided with an ESD.

**Explanation:** The error

```
The input line is too long
```

displays when upgrading Unwired Platform. If the system environment path is too long for the console window from which you run `start-unwired-server.bat`, an error occurs.

**Workaround:** The Sybase Unwired Platform installation directory path is too long. Shorten it, and retry. The default value is `C:\Sybase\UnwiredPlatform`.

### [Deprecated in 2.3]Client Application Hangs Waiting for Authentication

**Problem:** The client application hangs on the login screen, while waiting for authentication from Unwired Server. The application eventually times out and returns an authentication error: Failed to obtain `${licenseNumber}license(s)` for `SUP_BASECLIENT` feature from License Server `${server_id}`.

**Explanation 1:** This message is from SySam and indicates your site is using more client licences than your purchased SySam license allows.

**Solution:**

1. Check the Unwired Server log, and look for a checking out licenses error, for example: Failed to obtain 20 license(s) for SUP\_BASECLIENT feature from License Server <server\_id>.
2. Check your license file and find the license type. This file sample shows that only 10 licenses are allowed (CP=10):

```
INCREMENT SUP_BASECLIENT company_name 2011.03312 31-mar-2011
uncounted \
VENDOR_STRING=PE=EE;LT=CP HOSTID=00-1A-A0-36-D4-3F
ISSUER="CO=company_name;V=1.5;AS=A;MP=365;CP=10;" ISSUED=20-
jan-2010 NOTICE="For \
company_name. Internal Testing Use Only" SN=12727 TS_OK \
SIGN2="177C E3EA A4AF 4D7F F470 D299 E7B2 215F 8EBB 76A3 0717 \
FC2F ED9B A795 BE18 05FB 1871 6352 46D8 1FB7 EAE4 EDB1 CF70 \
CF6E 970A 6F51 2121 8044 57D9 B87F"
```

This example indicates there is a mismatch between the number of client licenses (20) and the number of Unwired Server licenses allowed (10), so Unwired Server does not authenticate.

3. Reconfigure the license by using `license.bat`, for example: `license.bat EE CP 10`
4. Contact Sybase to correct the license problem.

Explanation 2: This could occur when more device clients are activated and used, than the number of client licenses available.

Solution:

1. Check the Unwired Server log, and look for a checking out licenses error, for example: The number of registered devices has reached the maximum limit for your license..
2. Contact Sybase to purchase more licenses. Once purchased, run the `license.bat` file again.

Explanation 3: This could occur if you have not generated server or client licenses correctly.

Solution:

1. Check your `sup.properties` file to see your license configuration:

```
license.product.edition=EE
license.type=CP
client.licenses=0
```

In this example, 0 indicates there are no client licenses.

2. Check the *Installation Guides* for information about generating and downloading license files. See *Obtaining a License* in *Installation Guide for Runtime* or *Installation Guide for Sybase Mobile SDK*.
3. Contact Sybase to correct persistent licensing problems.

Explanation 3: This message is from SySam and indicates your site is using more client licences than your purchased SySam license allows.

## Troubleshoot the System

Solution:

1. Check your `sup.properties` file to see your license configuration:

```
license.product.edition=EE
license.type=CP
client.licenses=-1
```

The `client.licenses` parameter refers to the number of licensed devices. A value of `-1` indicates there is no limit to the number of devices.

2. Check the Unwired Server log, and look for a checking out licenses error, for example: Failed to obtain 2147483647 license(s) for SUP\_BASECLIENT feature from License Server 27001@mmdfwmap299.mm.local. This example indicates there is a mismatch between the number of client licenses and the number of Unwired Server licenses, so Unwired Server does not authenticate.
3. Check the license folder and find the license type. For example, you might find the type is ED (and not EE). This means the license configuration must be reset in order for Unwired Server to check out the right client licenses.
4. Contact Sybase to correct the license problem.

### Performance Issues if Unwired Server Not Installed on a 64-bit Machine

**Problem:** You observe performance issues when a high volume of client activity (synchronizations or logins) occurs on an Unwired Server running on a 32-bit Windows machine.

**Explanation:** Unwired Server is only supported on 64-bit machines for production deployments. The use of 32-bit machines do not provide a large number of TCP/IP ports by default. An indication of this problem are synchronization and login error messages in the client log.

**Solution:** Use a 64-bit Windows server installation for all work other than development testing.

### Unwired Server or RSOE Startup Problems

**Problem:** Problems, such as Unwired Server or RSOE failing to start, are encountered without any obvious error messages.

**Explanation 1:** This may occur if the Unwired Server host machine is running intrusion detection software, such as the McAfee Host Intrusion Prevention service. This may cause a connection from the CDB to the Unwired Server node to be flagged as an attack, and the CDB host to be added to the list of Blocked Hosts, preventing communication between Unwired Server and the CDB. The Unwired Server log reports connectivity problems with the CDB, and startup failure problems similar to:

```
YYYY-MM-DD HH:MM:SS.SSS ERROR MMS Thread-55
[com.sybase.djc.log.SystemExceptionLog]
com.sybase.djc.DataStoreException was thrown by method
```



```
com.sybase.djc.server.PartitionLock.updateLock(boolean)
\ncom.sybase.djc.DataStoreException: Connection Failed:
java.sql.SQLException: JZ006: Caught IOException:
java.net.SocketException: Software caused connection abort:
connect ...
```

**Workaround 1:** You may want to set up some exclusion rules to allow the required communication between the CDB and Unwired Server nodes. Use the following information to formulate the exclusion rules; rules may vary depending on the intrusion detection software used.

- **SQL Anywhere® to Unwired Server** – SQL Anywhere sends UDP packets on port 2638 in response to discovery probes. The JDBC/ODBC client (on Unwired Server) sends a UDP broadcast on 2638 outbound to all servers on the local network(s) with the name of the database service to which the client is trying to connect. If a SQL Anywhere hosting the given database sees that UDP broadcast request, it responds with a UDP targeted directly back to the host that sent the broadcast.

The installer also issues a 'dblocate' query, which receives UDP responses back from every SQL Anywhere on the local network. Unwired Server uses that to filter out < *cluster-name* > *\_clusterdb* responses to form a list of reachable Sybase Unwired Platform clusters to populate the dropdown list for subsequent installs where the user is adding a new Unwired Server to an existing cluster. If Sybase Unwired Platform is installed in a cluster, the MobiLink™ services also open connections among each other to coordinate activities of the MobiLink farm. Those connections are on the MobiLink listener ports.

- **Sybase Unwired Platform to CDB** – uses a JDBC connection through jConnect™ for JDBC™ driver. There are two kinds of protocols between Sybase Unwired Platform and CDB:
  - The MobiLink and CDB connection uses Command Sequence.
  - The jConnect for JDBC connection uses TDS.
- **RSOE to Relay Server** – uses either an HTTP and HTTPS connection.
- **Unwired Server to Unwired Server** – for administration communication uses:
  - Administration communication (MMS to MMS) uses IIOPS protocol by default.
  - Administration communication (MMS to Mobile Office service) uses IIOPS protocol by default.
- **Unwired Server to Advantage Database Server** – uses an OLE DB connection through ADO driver.

**Explanation 2:** If .Net is present when Unwired Platform is installed, and the .Net installation is corrupt, the Unwired Platform installer does not detect the problem, leaves the corrupt .Net in place and Unwired Server does not start.

**Workaround 2:** Uninstall both Unwired Platform and .Net, then reinstall Unwired Platform, letting the Unwired Platform installer install .Net..

## .NET Environment Problems Affect Unwired Server

**Problem:** When installing or upgrading Unwired Server, certain .NET applications which are required by Unwired Server, such as OBMO.exe, do not run.

**Explanation:** There is a problem with the .NET environment which prevents running any .NET applications on the Unwired Server. You see a message in the Event Viewer similar to: "The type initializer for System.Transactions.Diagnostics.DiagnosticTrace threw an exception."

**Solution:**

1. Uninstall all .NET versions.
2. Uninstall all VC ++ Runtime versions.
3. Restart the system.
4. Re-install Sybase Unwired Platform.

## Messaging Installer Completed With Errors

**Problem:** When upgrading to a newer software level, you are informed of errors.

```
****Error!!! Error!!!****
The Messaging installer completed with the following errors. These
errors will have to
be addressed before the services become functional.
Please refer to documentation for recovery process.
Database upgrade failed."
```

**Explanation:** This error happens when the messaging server database upgrade does not complete. The most likely reason for this is that the messaging server database had some lingering connections to a table while the upgrade was running. You can get more detailed information from the trace file located in SUP\_HOME\Servers\UnwiredServer\Logs\DBUpgrader.

If you find this syntax error in the DBUpgraderTrace.log, you need to configure the cache database (CDB) with the database option `quoted_identifier='ON'` for all database users:

```
20120918T234644.977|ERROR: Syntax error near 'ENCRYPTED' on line 19
20120918T234644.980|-- Caught unexpected exception: Error creating
database schema
20120918T234644.981|Got global error [Error creating database
schema]
```

**Solution:**

1. Skip this step if you did not find the syntax error in DBUpgraderTrace.log, otherwise configure the cache database (CDB) with the database option `quoted_identifier='ON'` for all database users.
  - a) Use the **dbisql** tool.

b) Run this SQL statement:

```
set option quoted_identifier = 'ON'
```

2. Stop all Unwired Server Unwired Server services in all the application nodes.
3. Restart the database services in the data tier node. This is to make sure the Messaging Database has no open connections.
4. From the primary server node, open a Windows command window and navigate to SUP\_HOME\Servers\MessagingServer\Bin.
5. Run the command **dbupgrader -v current** and make sure no error messages are displayed.
6. Start the Unwired Server Unwired Server service in the application node and verify it can start.

## Troubleshoot Deployment

---

Provides troubleshooting information for deployment problems.

**Note:** Before performing troubleshooting based on specific symptoms, collect information on the mobile application project. See *Troubleshooting > Troubleshooting Basics > Collecting Information > Mobile Application Project and Deployment Information*.

---

### Result Set Filter or Result Checker Class Files Not Deployed To Unwired Server

**Problem:** You deploy a mobile business object that includes either result set filters or result checkers to the master Unwired Server but the required class files did not get copied as expected.

**Solution:** Ensure that you include any JAR files that contain result set filters or result checkers during deployment using Sybase Unwired WorkSpace. The deployment wizard includes a Package User-defined Classes window that allows you to create and add JAR files to the deployment package.

To locate the name of the primary server, look at the server list in Sybase Control Center. The primary server in a cluster is labeled as "primary" in the left navigation pane.

### Modified Configuration Profile Changes Not Deployed

**Problem:** Modified data source connection information is not reflected when you deploy mobile business objects (MBOs).

**Explanation:** In Sybase Unwired WorkSpace, an MBO package is created from a database connection profile (for example, "PurchaseOrderDB") with particular connection parameters (for example, host and part of example1:9999), and then deployed to the Unwired Server.

During deployment, you create a Server connection (for example, "PurchaseOrderDB") in the Server Connection Mapping window, which creates a file in the Unwired Server at C :

## Troubleshoot the System

```
\Sybase\UnwiredPlatform\Servers\UnwiredServer\Repository
\Instance\com\sybase\djc\sql\DataSource
\default_PurchaseOrderDB.properties with connection information to the
database.
```

In Enterprise Explorer, you change the connection profile information for the connection profile "PurchaseOrderDB" (for example you change the host and port to example2:9999) and then try to redeploy the package to the Unwired Server. During deployment, when you select PurchaseOrderDB under Server connection in the Server Connection Mapping window, its connection properties are listed in the table, and still show the host and port, example1:9999.

This behavior occurs because the information is retrieved from Unwired Server, and does not consider the change from the Enterprise Explorer connection profile (example2:9999).

Solution 1:

1. Create a new "Server connection" during the deployment in Unwired WorkSpace, and the deployment wizard will fill in the new connection information.
2. Redeploy the MBO package from Sybase Unwired WorkSpace, using the newly created Unwired Server connection.

Solution 2:

1. In the Unwired Server installation directory, change to this folder:  

```
<UnwiredPlatform-InstallDir>\UnwiredPlatform\Servers
\UnwiredServer\Repository\Instance\com\sybase\djc\sql
\DataSource\

```
2. Open the `.properties` file and change the connection information.

## **NOCLOBBER Deployment Error**

Problem: After restoring the runtime database, an attempt to redeploy a package results in a deployment error.

```
ds_deploy_error: ----DataService deployment error: NOCLOBBER
```

Explanation: This could happen if the package is being deployed since the last database backup. The file system data corresponding to the package may be out-of-sync with the database contents related to the package.

Solution:

1. Navigate to: `SUP_HOME\Servers\UnwiredServer\deploy\sup`
2. Manually delete the package in the `\sup` folder.
3. Redeploy the package.

## Troubleshoot Synchronization

Provides troubleshooting information for problems that can occur with embedded synchronization components, such as the cache database (CDB), the remote device database, or the synchronization server.

### Synchronization Failed while Downloading Data

**Problem:** You receive a synchronization failure message while downloading data.

**Explanation:** This may indicate a deadlock problem with threads in the Unwired Server cache database (CDB) used to consolidate enterprise information system (EIS) and device client data. You can confirm this problem by reviewing the exception details and looking for this text in the Unwired Server log file:

```
[Sybase][ODBC Driver][SQL Anywhere]All threads are blocked
```

**Solution:** Set the number of concurrent threads for the server CDB to a value that is higher than the number of work threads configured for the synchronization server. To do this, click **Server Configuration** in the left navigation pane of Sybase Control Center for Unwired Server. Change the database server's thread count property as documented in *System Administration*.

### Multi-threaded Delete Failures Not Logged

**Problem:** If you have multiple requests from different devices trying to delete the same record at the same time, and a thread successfully deletes the record, but a second thread fails because that record is now removed from the server, no failure message appears in the Unwired Server log.

**Solution:** Use the old value argument in Unwired Workspace for optimistic concurrency control (OCC). See *Sybase Unwired Workspace - Mobile Business Object Development > Develop > Working with Mobile Business Objects > Mobile Business Object Data Properties > Old Value Argument*.

### Device Synchronization Returns no Data

**Problem:** Improper syntax in generated code calls causes device synchronization to return no data.

**Explanation:** This may originate from a misconception about the syntax of generated code calls. In this code example, the deployment\_unit file used to autogenerate the class objects contains this section:

```
<synchronization-parameter name="city" type="string?"/>
```

## Troubleshoot the System

This indicates that the class object created from the XML requires "city" to be specified prior to syncing with the object.

Solution: Research the mobile business object (MBO) design in Eclipse.

1. Make sure that the synchronization parameter is mapped to an attribute:
  - a) In MBO properties, from the Synchronization tab, select a sync parameter.
  - b) Select an attribute for the "Mapped to" column.
2. After making the change, redeploy the Mobile Application project and regenerate the Object API code, and resynchronize the MBO on the device.

## **Errors When Reusing Primary Keys**

Problem: Reusing an enterprise information system (EIS) primary key generates an error. If you insert a record that has the same primary key that has been deleted, the Unwired Server fails to retrieve the inserted data.

Explanation: Make sure the EIS does not reuse primary keys. For example, if a record with primary key value 1 is deleted, make sure records inserted do not use 1 as the primary key value.

For a database EIS, you can define the primary key column as auto-increment, so when a new record is inserted, the primary key is always assigned a new value.

## **Troubleshoot Runtime**

Provides troubleshooting information for runtime processes, for example application execution.

## **Errors if CUD operations are not defined for an MBO**

Problem: If you do not define create, update, or delete (CUD) operations for your MBOs in Unwired WorkSpace, you get exceptions when calling the CUD of that MBO in client object API code.

For example:

1. Use DCN upsert to modify MBO data and push the data to the device.
2. The device application user modifies the data on the device.
3. If no CUD operations were defined for the MBO, the application code gets an exception when it calls [mbo save], [mbo delete], [mbo create], or [mbo update].

This is expected behavior for all supported object APIs.

Solution: Define MBO operations if you plan on calling them within client object API code.

## **Troubleshoot the Unwired Server Cache**

---

Provides troubleshooting information for problems that can occur with the cache database (CDB).

### **Primary Key Record Missing**

**Problem:** Two records with similar primary key names should exist but only one appears in the cache. Unwired WorkSpace returns an error that the primary key attributes only differ in case.

**Solution:** Check to see whether your datasource is configured as a case insensitive repository. Unwired Platform requires that the datasource be case insensitive. This may require that you change at least one primary key name so they use different characters. Without these changes the first primary key data is overwritten by the second, and this issue will be propagated to client databases when the user synchronizes.

### **Data Caching Problems**

**Problem:** A data caching problem is detected, or data caching is not operating as expected.

Symptoms may include:

- User 1 downloads the data of User 2 – this might happen if only one version of the data is cached because:
  - Two different device users are using different runtime credentials to access data from the backend, and the backend returns data based on the authenticated user.
  - The coherence window is larger than the interval in which the users are syncing.
- All users end up with data on the device that was originally cached by demand of a different device – this might happen if a customized download cursor is used by multiple CSI users using different load argument values (via different sync parameters).
- If a backend returns data based on an argument that is not filtered, because it is not an attribute of the MBO, all users will download that data by design. If you want to prevent this from happening, propagate the load argument into an attribute, so that it can be used as a filter later.

**Solution:** Implement data partitioning in your MBO design. To enable partitioning of data by user for an MBO located in a cache group with a on-demand refresh policy:

1. Model an additional synchronization parameter and load argument onto the MBO that requires partitioning, and fill the load argument with the modeled sync parameter. The sync parameter should be filled with the system personalization key "username".
2. Propagate the artificial load argument into an artificial attribute of the MBO which must be marked as part of the composite primary key of that MBO.

## Troubleshoot the System

3. Map the synchronization parameter to the attribute into which the load argument was propagated.

### **No Data or Images Appear on the Device**

Problem: No data or images appear on the mobile device.

Explanation 1: This may indicate that operation parameter datatypes (`int`, `decimal`, `image`, and so on) are mapped incorrectly. You map parameter datatypes to strings from the Mobile Application Diagram when you create an MBO operation.

Solution:

1. From the Mobile Application Diagram, double-click the operation to show in the Properties View.
2. Select the **Parameters** tab.
3. Select a datatype in the first **Datatype** column that matches the one in the second Datatype column. For the image type, you need to provide binary type, and the size of the binary.

Explanation 2: The synchronization result may be empty if these rules are not followed:

- When the backend is a SQL Anywhere database, and using the `CHAR NULLABLE` column as a sync parameter, you do not need to pad the value with spaces. When using the `CHAR NOT NULLABLE` column as a sync parameter, you need to pad the value with spaces.
- When the backend is a non-SQL Anywhere database, and using the `CHAR` column as a sync parameter, you always need to pad the value with spaces.

Solution: Re-evaluate the synchronization parameter value, and make modifications using these rules as a guide.

Explanation 3: This may indicate that a new cache database (CDB) in an existing SQL Anywhere environment needs trailing spaces enabled to perform proper comparison by ignoring the trailing blanks. When device client provides a synchronization parameter value, it must include trailing spaces. UTF8 collations are also a requirement for Unwired Platform.

Solution: See *System Administration for Sybase Unwired Platform > Environment Setup > Database Setup > Initializing a New Cache Database* for information about setting up the environment properly.

## **Troubleshoot Security, Logins, and Authentication**

Provides troubleshooting information for security, login, and authentication problems.

### **Cannot Log in to Unwired Server**

Problem: Log-in issues prevent Unwired Server administration.

Solution 1: You may be expecting single sign-on capability, when the same security mechanism has not been configured for both Unwired Server and Sybase Control Center. To



ensure the same credentials are used in all components, Sybase strongly encourages you to configure a common security provider for Unwired Server, Sybase Control Center, and Afaria. LDAP and the native operating system (OS) are the only providers that can be used by all Unwired Platform components.

Solution 2: You may have implemented the security provider or server configuration incorrectly, and as a result, cannot log in to Unwired Server. This typically requires that you manually edit a combination of the embedded properties files to correct the problem. Only manually edit the files under the advisement of Sybase technical support.

## Troubleshooting Single Sign-On

Provides troubleshooting information for problems that can occur when implementing single sign-on (SSO) for SAP enterprise information systems (EIS).

- **When testing, the error `No suitable SAP user found for X.509-client certificate` is logged on the client and Unwired Server when testing SSO with an X.509 certificate on a mobile application client** – during certificate generation, make sure that the user name under which the `dbsrv12.exe` process (Unwired Server) runs, is the same as the user name who generates the credential (`cred_v2`) file. For example, in the Processes tab of Windows Task Manager, the User name under which `dbsrv12.exe` runs is SYSTEM. In this case a SYSTEM user must also generate the certificate.

See this SAP Community Network link <http://forums.sdn.sap.com/thread.jspa?threadID=1756876>.

- **When testing, the error `RFC_ERROR_LOGON_FAILURE: No suitable SAP user found for X.509-client certificate` is logged on the client and Unwired Server when testing SSO with an X.509 certificate on a mobile application client.** –
  1. If using X.509 certificate authentication, remove the JCo properties `jco.client.passwd` (Logon User) and `jco.client.user` (Password) defined for the SAP connection profile in Sybase Control Center.
  2. Regenerate and redeploy the Workflow application and associated MBOs.

**Note:** You need to remove the SAP connection properties from Unwired WorkSpace.

- **Application login failures, authentication failures, and operation replay failure error messages** – any of these failures may be caused by the user's password changing in the SAP server. In some cases you may need to create new connection profiles.

For security reasons, the exact nature of the login failure is not returned to the client. Look at the Unwired Server log for details.

- **Using the `createcert` utility to create X.509 certificates to use in SSO testing** – Unwired Platform includes a SQL Anywhere installation that includes the `createcert` utility for generating X.509 certificates.

## Troubleshoot the System

See *System Administration > System Reference > Command Line Utilities > Certificate and Key Management Utilities > Certificate Creation (createcert) Utility* or the SQL Anywhere documentation for details.

- **Unexpected application behavior when you change credentials in an application** – when you change a password in the client application, it changes only the password which is sent to the SAP server. To change the password that the SAP server accepts, change it at the EIS.

You can import any valid certificate into the device's certificate store, but if the SAP server rejects it, an error is returned to the client.

- **SAP portal does not set SSO2 logon tickets** – you need a note applied to the portal to get the HTTP challenge pop-up.

Ask the basis team to apply note 1250795 to the portal server. See <https://websmp230.sap-ag.de/sap/support/notes/1250795> (requires SAP domain login).

## Server Key Verification Failed

Problem: You receive a Server Key Verification Failed (584) error.

Explanation: Once an application is registered with an Unwired Server, a bond is established. This message may indicate that the application is attempting to connect to a server other than the original.

Solution: If you want your application to use a different Unwired Server, you have the following two options:

### *Option 1*

Uninstall, re-install, and re-register the application.

### *Option 2*

1. Unregister the application from the original Unwired Server (via `unregisterApplication`).
2. Delete the application's database on the device.
3. Register the application with the new Unwired Server (via `registerApplication`).

See *Deleting the Database and Unregistering the Application* in *Developer Guide: <Device Platform> Object API Applications* for platform-specific syntax.

## Troubleshoot EIS Connections

Provides troubleshooting information for enterprise information system (EIS) connection problems.

## DOE-C or Messaging Connection Test Errors

Problem: Test of connection properties for messaging or SAP Data Orchestration Engine connector (DOE-C) package fails.

Explanation: The table below lists several common error messages and resolutions.

Error	Package type	Likely cause and resolution
Ping failed	Messaging DOE-C	Server at Service Address is not online. Start up that server.  Alternatively, this may indicate the wrong ESDMA Interface version is in use. The minimum version required is "14."
Unauthorized	Messaging DOE-C	Username or Password are incorrect. Correct the credentials.
Read timeout	DOE-C	Listener URL is incorrect. Remove and redeploy ESDMA package.

## Data Source Performance Errors

Problem: Response time for client synchronization degrades, or Unwired Server reports timeout errors while obtaining connections.

Explanation: You may see playback and timeout exceptions reported when some queries are executed:

```
ResourceMonitorTimeoutException: data
source:<datasource_name>).
```

You may see errors that indicate client synchronization performance is degraded:

```
Client receives an error code -85
(SQLC_COMMUNICATIONS_ERROR)).
```

These errors indicate that the connection pool size is too small for your user base, and that resources are not managed sufficiently.

Solution: Increase the connection pool size with the `maxPoolSize` property. This value change increases the number of connection pools available to users on your network. You can also increase the value for `maxWaitTime` to increase the timeout value used by the data source.

You set this in the data source connection. In Sybase Control Center, navigate to **Domains > <Domain-> > Connections**, then select the connection and click **Properties**. The property value can be modified or added as needed.

## Troubleshoot the System

You can modify connection properties either from Sybase Control Center by clicking **Connections** on the navigation pane, or you can manually modify the ANT file in `<UnwiredPlatform_InstallDir>\Unwired Platform\Servers\UnwiredServer\config\default-data-sources.xml`.

After changing this file, run `configure.bat default-data-sources` to implement changes.

---

**Note:** Run `configure.bat default-data-sources` only if you manually modified `default-data-sources.xml`.

---

## SAP Issues

Provides information about issues that may arise when connecting to an SAP® enterprise information system (EIS).

### Verifying SAP Prerequisites

You can avoid many SAP issues by meeting the basic prerequisites required to integrate your SAP system with the Unwired Platform development environment.

#### *Server prerequisites*

Check the following for each Unwired Server in the production configuration:

- Verify that the SAP version to which you are connecting is supported. Check *Supported Hardware and Software* for supported versions.
- On Windows 7, Windows Vista, and Windows Server 2008, verify you have the correct versions of Microsoft `msvcr71.dll` and `msvc71.dll` files, and that they are installed in the `WINDOWS_HOME\System32` (32-bit) or `WINDOWS_HOME\SystemWow64` (64-bit) directory. See *System Administration* for the topic *Preparing Unwired Server to Connect to SAP using Java Connectors*.
- If you are operating in a clustered production environment, you must also enable SAP mobile business objects to connect to an SAP R/3 system that uses a router. See *Preparing Unwired Server to Connect to SAP using Java Connectors* in *Landscape Design and Integration*.
- If you are using a connection pool, make sure it is configured correctly. Search *System Administration* for the topic *EIS Data Source Connection Properties Reference*.
- If you are using the SAP Data Orchestration Engine Connector (DOE-C), make sure it is configured correctly. Search *System Administration* for the topic *EIS Data Source Connection Properties Reference* and its SAP-related sub-topics.
- If you are using Single Sign-on (SSO), make sure you are configured correctly for the type of credential you are using—X.509 certificate, or SSO2 Token. Search *Security* for the *Enabling Single Sign-on* topics.

#### *Development prerequisites*

Check the following for each Unwired Server in the development environment:

- Verify that the SAP version to which you are connecting is supported. Check *Supported Hardware and Software* for supported versions.
- Unwired WorkSpace – on Windows 7, Windows Vista, and Windows Server 2008, verify you have the correct versions of the Microsoft `msvcr71.dll` and `msvcrt71.dll` files, and that they are installed in either the `<WINDOWS_HOME>\System32(32-bit)` or `<WINDOWS_HOME>\SysWOW64 (64-bit)` directory.

## Troubleshoot Servers

Provides troubleshooting information for servers used in the Unwired Platform configuration.

### Unwired Server

Provides troubleshooting information for Unwired Server.

#### Troubleshooting Unwired Server Performance Issues

Problem: Unwired Server exhibits performance issues.

Explanation: The performance configuration may not be properly tuned to optimize system operation.

Solution: Ensure that all applicable performance-related settings are properly configured.

- Java heap size – when the garbage collection log indicates high activity and the JVM memory allocation pool heap size remains close to the maximum, you may need to reconfigure the maximum and minimum heap sizes. You can enable garbage collection tracing by setting the "JVM options" property. Set appropriate maximum and minimum heap sizes in the **Server Configuration** node (Performance Configuration tab) of Sybase Control Center. The default minimum heap size is 512MB and the default maximum heap size is 2048MB. See *Configuring System Performance Properties*.
- Java thread stack size – if you see a `StackOverflowError`, you may need to increase the default stack size from 400KB. Set an appropriate Java thread stack size in the **Server Configuration** node of Sybase Control Center. Alternatively, you can also decrease the stack size to save more memory. See *Configuring System Performance Properties*.
- Messaging synchronization (MBS) – if there are large numbers of MBS clients and response times or queue counts are high, consider increasing the inbound and outbound messaging queue counts. These numbers control the number of message queues used for incoming and outgoing messages between the messaging-based synchronization application and the server. If you increase either of these values, ensure you also increase the `maxThreads` property of in the `<hostname>_iiop1.properties` file, located in `SUP_HOME\Servers\UnwiredServer\Repository\Instance\com\sybase\djc\server\SocketListener\`.

View messaging queue statistics in the **Monitoring** node of Sybase Control Center to calculate the appropriate values and enter these in the **Server Configuration** node of Sybase Control Center. See *Configuring System Performance Properties*.

- Replication synchronization – if replication synchronization client application performance does not meet the desired level and the CPU or memory settings are not causing this issue, consider increasing the synchronization thread count and cache size for the replication synchronization server. Choose an appropriate synchronization cache size and thread count in the **Server Configuration** node of Sybase Control Center. See *Configuring Replication-Based Synchronization Properties*.
- Cache database (CDB) – if the CDB server is slow, consider increasing the CDB thread count to allow for more parallel threads to serve incoming requests from the Unwired Server nodes. See *Changing the Cache Database Server Thread Count*. Also consider tuning the number of client connections that are saved in the CDB connection pool. Configure this in the 'default' domain by modifying the 'default' data source pool size property in Sybase Control Center. Repeat this change on all nodes in the cluster.
- Data change notification (DCN) listener threads – if there are a large number of concurrent requests and processing times for those requests do not meet requirements, consider increasing the number of DCN listener threads. Set an appropriate number of listener threads in the **Server Configuration** node of Sybase Control Center. See *Configuring Communication Port Properties in Sybase Control Center for Sybase Unwired Platform*.
- Server host memory – if you are running Unwired Server on a host with 4GB of memory, by default, 2GB of memory is allocated to applications, such as Unwired Server. If performance issues persist after you have adjusted the aforementioned settings, consider starting Windows with a "/3GB" switch. See [http://technet.microsoft.com/en-us/library/bb124810\(EXCHG.65\).aspx](http://technet.microsoft.com/en-us/library/bb124810(EXCHG.65).aspx).
- Platform connections – for JDBC connections, consider setting the maximum connection pool size to restrict the size of the pool. You can set this value in Sybase Control Center. See *Data Source Performance Errors*.
- Sybase Unwired Platform connection settings to the backend EIS – for JDBC connections to the backend database, you can add options such as `maxPoolSize=<no. of pool size>` to the connection cache property files, located in `SUP_HOME\Servers\UnwiredServer\Repository\Instance\com\sybase\djc\sql\DataSource\*.properties`.

For expanded performance tuning details, especially in regards to synchronization performance, see *Performance Tuning in System Administration*.

### **Troubleshooting Device Tracking in a Cluster**

**Problem:** The server logs for primary and secondary servers list several -10117 stream errors (also known as pull-based notification for replication synchronization).

**Solution:** The Unwired Server cache database has issues with tracking devices and light weight polling (LWP) in a clustered environment. You must disable device tracking when LWP is used by reconfiguring all servers in your cluster.

### **Server Configuration Changes Not Reflected after Server Restart**

**Problem:** You do not see your server configuration changes reflected after restarting the Unwired Server.

**Explanation:** After changing the configuration using Sybase Control Center (for example, to use HTTPS for synchronization), the administrator did not use Sybase Control Center to restart the Unwired Server, nor use `start-unwired-server.bat` to start and restart the Unwired Server. The administrator only restarted the Sybase Unwired Platform server service using the Windows Services Manager. This method does not make the Sybase Control Center changes take effect.

**Solution:** Issue a restart from Sybase Control Center or use `stop-unwired-server.bat` then `start-unwired-server.bat` to restart the Unwired Server.

### **Unwired Server Startup Problems After Changing System Time**

**Problem:** Encountered problems starting up Unwired Server after changing the system time.

**Explanation:** This can happen if you change the system time to an earlier time setting. If you check the Unwired Server logs you may see messages similar to:

```
YYYY-MM-DD HH:MM:SS.SSS ERROR MMS Thread-8
[com.sybase.sup.server.lm.LicenseManager] msgId: 131160, message:
System clock has been set back.
```

1. Revert the machine to its original time.
2. Uninstall Sybase Unwired Platform.
3. Reset the time if desired.
4. Reinstall Sybase Unwired Platform.

### **Process of Stopping Unwired Server Service Hangs**

**Problem:** When stopping the Windows service for Unwired Server, very infrequently the process hangs. This problem is independent of the means used to stop the service..

**Explanation:** When this happens, the process tree for the service for the Unwired Server service is no longer intact, so you cannot use the Task Manager End Process Tree option to complete the process of stopping the service.

#### **Workaround:**

1. Open Windows Task Manager on the system where you want to stop the Unwired Server service.
2. Activate the **Process** tab and click the **Image Name** column heading to sort the names in alpha order.
3. For each of the following processes that are present in the list, right-click the process and select **End Process**.

## Troubleshoot the System

- AdminWebservices.exe
- AMPService.exe
- Dbsrv12.exe
- Dbsrv11.exe
- Java.exe
- JmsBridge.exe
- LBManager.exe
- mlsrv12.exe
- MlsrvWrapper.exe
- OBMO.exe
- OBServiceManager.exe
- Rsoe.exe
- Sccservice.exe
- wmiprvse.exe

4. Go to the Windows Services panel and verify that the Unwired Server service is stopped.

### **Version Mismatch Detection**

The `isCompatible` method prevents version mismatch between incompatible versions of Unwired WorkSpace and Unwired Server.

Examples include:

- A later version of Unwired WorkSpace connecting to an earlier version of Unwired Server.
- An earlier version of Unwired WorkSpace connecting to a later version of Unwired Server.

If the two are incompatible, an error message is generated. For this version of Unwired Platform, when Unwired WorkSpace attempts to connect to an older Unwired Server version from a connection profile, the error "UnsupportedOperationException" displays and connection fails.

```
boolean isCompatible(string server_host, int iiop_port, int client_version)
```

Unwired Server responds with true or false based on the version identifier passed from the client and is checked for support of that version. This method works with any Unwired Server in a cluster, without the need to specify the host and port of the primary.

### **Cannot Ping Unwired Server**

Problem: Cannot ping the Unwired Server from the Unwired WorkSpace development environment.

Explanation: Verify that Unwired Server is up and Sybase Unwired Platform services are running.

Solution: Use Sybase Control Center to verify Unwired Server is up and running. Use the Control Panel to verify Sybase Unwired Platform services are running, and restart any that are not running. See:



- *Sybase Unwired Platform Windows Services in System Administration*
- *Starting Required Services in Landscape Design and Integration*

### **ODBC Driver Errors**

Problem: ODBC driver errors appear when you try to connect to a database from Sybase Unwired Platform installed on a 64-bit operating system.

This can happen if the registry has been changed, so the ODBC driver cannot be found. For example, installing a standalone version of SQL Anywhere may cause registry changes.

Solution: Restore the registry settings for the ODBC driver. From the Command Prompts window, execute these commands:

```
regsvr32.exe
```

```
SUP_HOME\Servers\SQLAnywhere12\bin32\dbodbc12.dll
```

### **Unwired Server Service Startup Failure Post-Install**

Problem: The installer creates a service that is defined by the user performing the installation. When Windows forces a domain password change, the Unwired Server service fails to start, because the former domain password is still used.

Solution: Change the password for the service by:

1. In the Windows Control Panel, open the Services pane.
2. Right-click Unwired Server service, and choose **Properties**.
3. In the **Log On** tab, configure properties for the host computers user account and ensure the newly changed domain password is used.
4. Click **Ok**.
5. Right-click Unwired Server service, and choose **Restart**.

### **Unwired Server Fails to Start**

Problem: The Unwired Server fails to start even though the Unwired Server service has started.

Explanation: The Unwired Server service manages multiple child processes, and its running status does not indicate that the actual system is fully functional. Among these child processes, the most important are `OBMO.exe` and `mlsrv12.exe`. If none of these are continuously running or multiple instances are running, it will signify a problem. The most common cause is an environment issue (for example, missing files, cannot connect to CDB or Messaging DB) that prevents these processes from starting.

Solution 1: First check the Event Viewer application log to see if any errors are reported by Unwired Server. Address any reported issues and restart the service. If this does not solve the problem, check the `mlsrv_err.log` under the logs directory. This log reports issues with the `mlsrv` process during startup. There are additional log files in the `logs\OBServiceManager` folder that may have information on the Unwired Server startup issues.

## Troubleshoot the System

If the problem involves having multiple instances of child processes running, the best way to recover is by restarting the computer.

If for any reason, a forced stop of the Unwired Server is needed, you must terminate the entire process tree under the process "OBServiceManager."

**Solution 2:** Check for a JVM creation fails error message in the `mlsrv_err.log`. This message typically indicates that the JVM maximum heap size is too large for the hardware that hosts Unwired Server. Use Sybase Control Center to change the current value to one that is smaller and restart the host node in question. For details, see *Configuring Unwired Server Performance Properties*. If you need guidance on choosing an appropriate value, see *Unwired Server Replication Tuning Reference* in *System Administration*.

### **Unwired Server Does Not Start When JVM Heap Value Set Larger than Upper Limit**

**Problem:** Unwired Server does not start properly.

**Explanation:** If Unwired Server does not start, check the Unwired Server log file for this message:

```
Could not reserve enough space for object heap
```

This message indicates that the maximum JVM heap size for Unwired Server is larger than the upper limit of the JVM heap size in Sybase Control Center.

**Workaround:** Manually change the heap size in the `mlsrv_ext.ini` file located in `SUP_HOME\Servers\UnwiredServer\bin`. For example, to `-Xmx512M`.

### **Unwired Server Fails to Start Due to Incorrect JVM Settings**

**Problem:** Unwired Server fails to start because of incorrect JVM property settings.

The JVM maximum heap size is set too low and/or a JVM option is missing a space.

**Solution:** To add a space to a JVM option:

1. Edit the `SUP_HOME\Servers\UnwiredServer\Repository\Instance\com\sybase\sup\server\SUPServer\sup.properties` file and search for `DJC_JVM_USEROPTIONS`.
2. Add a space before the appropriate option, for example, `-Dcom.sap.solutionmanager.url=<SolutionManager_URL>`.
3. Start the Unwired Server by doing one of the following:
  - Clicking the **Start Sybase Unwired Platform Services** desktop shortcut.
  - Run the `start-unwired-server.bat` from the command prompt.

**Solution:** To increase the maximum heap size:

1. Edit the `SUP_HOME\Servers\UnwiredServer\Repository\Instance\com\sybase\sup\server\SUPServer\sup.properties` file and search for `DJC_JVM_MAXHEAP`.
2. Increase this value. The default is 2048MB. Servers with 64-bit operating systems can be set much higher, depending on the available RAM.

---

**Note:** For servers with 32-bit operating systems, the `MAXHEAP` value is limited to 768MB. Some machines will have a maximum that is less than that, for example, 512MB. If the `MAXHEAP` is set too high, errors are reported in the `SUP_HOME\Servers\UnwiredServer\logs\mlsrv_err.log` file.

---

3. Start the Unwired Server by doing one of the following:
  - Click the **Start Sybase Unwired Platform Services** desktop shortcut.
  - Run `start-unwired-server.bat` from the command prompt

---

**Note:** To make sure that Unwired Server is reconfigured with the new heap size, make sure an empty `sup.cff` file exists in the `SUP_HOME\Servers\UnwiredServer\bin\private` folder. This file causes Unwired Server to reconfigure when it starts up using the `BAT` file. If the file does not exist, create it. You can copy `installdate.cff` and rename it to `sup.cff`.

---

### **Key or Keystore Messages Received in Server Log**

**Problem:** An Unwired Server error message regarding keystore tampering prevents Sybase Control Center from connect to Unwired Server.

One of two messages may be displayed in the server log console of Sybase Control Center when keystore, truststore, and private key passwords are not correctly implemented in the runtime:

- **Message for keystore and truststore password issues: –**

```
SUP server error message:
2012-03-13 13:32:02.535 CONSOLE MMS Thread-58
[com.sybase.sup.server.Console] java.io.IOException: Keystore was
tampered with, or password was incorrect
```

- **Message private key password issues: –**

```
WARN MMS Thread-5 [com.sybase.djc.server.jetty.WebServer] Failed
to
start Web Server: 'java.io.IOException: Cannot recover key'
```

**Solution:**

- **For non-SSL management ports of 2000 –** Follow these steps:
  1. Ensure the Unwired Server is a registered resource in Sybase Control Center, and that the regular, non-SSL port is configured. See *Adding or Updating Unwired Server Registration Properties* in *Sybase Control Center for Sybase Unwired Platform*.
  2. Ensure that the general server configuration uses the correct keystore and truststore passwords. If the passwords were changed with the `keytool` utility, but not updated in

Sybase Control Center, the connection fails with the documented error message. In this case, change the password here, and restart Unwired Platform services using the Windows Control Panel services tool. For details, see *Changing Keystore and Truststore Passwords* in the *Security* guide.

- **When the secure management port of 2001 is the only port enabled** – Then just ensure that the general server configuration uses the correct keystore and truststore passwords as documented in the second step for regular ports (2000).

### **Unwired Server Port is Not Accessible from a Remote Machine**

**Problem:** A port in the Sybase Unwired Platform environment is not accessible from a remote machine.

**Explanation:** This may indicate the machine is configured with more than one Ethernet adapters/IP addresses. Consequently a service port on Unwired Server may be configured with an IP address that is not accessible from the remote machine.

**Solution:**

1. From the command line, issue **ipconfig** to find out more information.

```
Windows IP Configuration
Ethernet adapter Local Area Connection* 11:
 Media State : Media disconnected
 Connection-specific DNS Suffix . :
Ethernet adapter Local Area Connection 2:
 Connection-specific DNS Suffix . : name.com
 Link-local IPv6 Address : xxxx:xxxx:xxxx:xxxx
 IPv4 Address. : xxx.xxx.xxx.xxx
 Subnet Mask : 255.255.255.0
 Default Gateway : xxx.xxx.xxx.xxx
Ethernet adapter Local Area Connection:
 Connection-specific DNS Suffix . :
 Link-local IPv6 Address : xxxx:xxxx:xxxx:xxxx
 Autoconfiguration IPv4 Address. . : xxx.xxx.xxx.xxx
 Subnet Mask : 255.255.0.0
 Default Gateway :
Tunnel adapter Local Area Connection* 8:
 Media State : Media disconnected
 Connection-specific DNS Suffix . : name.com
Tunnel adapter Local Area Connection* 9:
```

2. Check for multiple Ethernet adapters and IP addresses assigned to the machine.
3. Disable any unidentified Ethernet networks.
4. Retry connecting to the port from the remote machine.

### **Messaging Server**

Provides troubleshooting information for the embedded Unwired Server messaging service.

### Unable to Register the MobiLink Server

Problem: Starting Unwired Server causes registration problems for Messaging Server.

The Unwired Server error log displays an entry similar to:

```

2010-07-27 09:37:31.205 ERROR Mobilink Thread-146
 [com.sybase.ml.sup.Logger] [-10002] Cache database server or
ODBC
 error: ODBC: [Sybase][ODBC Driver][SQL Anywhere]Index 'ml_server
UNIQUE
 (name)' for table 'ml_server' would not be unique (ODBC State =
23000,
 Native error code = -196)
2010-07-27 09:37:31.205 ERROR Mobilink Thread-146
 [com.sybase.ml.sup.Logger] [-10294] Unable to register the
MobiLink
 server with name '<serverName>'

```

Explanation: When Unwired Server runs in a shared server farm, every member in the farm must have a unique registration through the cache database table **ml\_server**. If Unwired Server is not stopped gracefully, the row for the server is not removed from the **ml\_server** table. This could happen if Unwired Server is shut down abruptly while a mobile client is synchronizing. When Unwired Server is restarted, the Messaging Server registration is not unique, and cannot be completed.

Solution:

1. Connect to the cache database using **dbisql**.
2. Manually remove the row corresponding to your server. An example row:

```

server_id,name,version,connection_info,instance_key,start_time,li
veness 9,'ML Server
name','100.0','http(host=<HostName>;port=2480)',
0xad53f35e04086778623aa56f32728a313f1972b68b8363c86b501144af7b320
6,'2010-07-27 14:54:10.298','2010-07-27 18:29:13.866'

```

3. If the problem persists, try restarting Unwired Server.

## Relay Server

Provides troubleshooting information for Relay Server or Sybase Hosted Relay Server.

### Relay Server Does Not Support IIS Application Pooling

Problem: In an environment with Relay Server and Sybase Unwired Platform, Relay Server runs for a while and then crashes and requires a restart.

Explanation: Relay Server does not support IIS application pooling. This is documented in *System Administration > Relay Server Clusters > Creating a New Relay Server Cluster > Enabling Application Request Routing for IIS Relay Server Hosts > Configuring the Relay Server IIS Host for Load Balancing*.

## Troubleshoot the System

Solution: Check the Relay Server log for a message similar to the following: 'Redundant outbound enabler connection for backend server 'S0' in backend farm 'farm' was ignored.'

Also, check the Windows System Event Viewer for a warning on the Application Pool of the Relay Server.

Turn off application pooling for RelayServer in IIS, and stop and restart IIS.

### **Failed to Attach to Relay Server Shared Memory**

Problem: Failed to attach to the relay server with the message "<rshost> failed to attach to Relay Server 'ias\_relay\_server\_host' shared memory."

Explanation: This message may appear if you try to update Relay Server configuration using the command `rshost.exe -u -qc -f rs.config`.

Solution: Remove the `-u` option, since the server is not up and running. The `-u` option updates the configuration of a running relay server.

### **HTTP\_BAD\_STATUS\_CODE Error**

Problem: The HTTP\_BAD\_STATUS\_CODE error appears on the console or in the log files.

Solution 1 (iOS):

1. For iOS devices, navigate to the Settings application on the device.
2. Click the corresponding application tab.
3. Confirm that the connection information is correct

Solution 2 (Windows Mobile):

1. For Windows Mobile devices, check the connection parameters in the device client application code to make sure the relay server connection configuration is correct.
2. Confirm that the `rshost.exe` is running under the system user ID.

### **SOCKET\_HOST\_NAME\_NOT\_FOUND Error**

Problem: The SOCKET\_HOST\_NAME\_NOT\_FOUND 10050 error appears on the console or in the log files.

Solution 1 (iOS):

1. Navigate to the Settings application on the device.
2. Click on the corresponding application tab.
3. Confirm that the connection information is correct.

Solution 2 (Windows Mobile):

1. For Windows Mobile, if you are using an emulator for testing, confirm that you have synchronized the emulator first.

2. Confirm that `dblgen11.dll` is in your `C:\Inetpub\wwwroot\ias_relay_server\server\` folder.
3. Confirm that the RSOE version is less than or equal to the relay server version.

### **Connection Dropped Due to Lack of Network Activity**

**Problem:** The error message `Connection was dropped due to lack of network activity` appears on the console or in the log files.

**Explanation:** This message indicates that the relay server outbound enabler (RSOE) sent a PING to Unwired Server synchronization service to determine whether it is active.

**Solution:** This message is informational and can be ignored. If you want to suppress the message, you can remove the `-v+` switch from the Unwired Server.

### **Sybase Hosted Relay Server**

Provides troubleshooting information for Sybase Hosted Relay Server.

## **Troubleshoot Clusters**

---

Provides troubleshooting information for cluster problems related to Sybase Unwired Platform.

### **Advantage Database Server Cannot Authenticate**

**Problem:** After installing Unwired Platform data tier servers in a failover cluster, an error message appears in the event viewer.

One or more of the following messages may appear:

- 7078: The Advantage Database Server cannot authenticate the user. Make sure the name and password are correct. `axServerConnect AdsConnect.`
- 5004: Either ACE could not find the specified file, or you do not have sufficient rights to access the file. The data dictionary file is not found on the server. `AdsConnect.`

**Explanation:** The message indicates the Unwired Server messaging service cannot locate the shared data folder, on the shared cluster storage that houses the database files, or cannot access the file.

**Solution:**

1. On a data tier host, check the following registry key value to find the location of the `adsserver.ini` file:  
`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Advantage\Configuration\TPS_LOGS`

## Troubleshoot the System

The default value is: C : \

2. If the `adserver.ini` file does not exist, create it in the location identified in the registry key value, on each data tier host.
3. Add the following lines to the `adserver.ini` file on each data tier host, to identify the local path to the shared data folder:

```
[ServerAliases]
share_name=local_path
```

where `share_name` is the file share resource group (or Client Access Point).

In this example, `supData` is the Client Access Point, and `F : \` is the local path on each data tier server host.

```
[ServerAliases]
supData=F:\
```

4. Restart the Advantage Database Server service (cluster resource). See *System Administration > System Reference > Unwired Platform Windows Services > Starting and Stopping Unwired Platform Server Services*.

## **Nodes in Cluster Do Not Start Because of JVM Heap Space**

**Problem:** Nodes in an Unwired Server cluster do not start properly after installing the Sybase SAP® Data Orchestration Engine Connector (DOE-C).

**Explanation:** If after installing DOE-C, Unwired Server cluster nodes do not start, check the Unwired Server log file for this message:

```
java.lang.OutOfMemoryError: GC overhead limit exceeded
```

.

**Workaround:** Add more JVM heap space to the affected Unwired Server hosts. See *Administration Workbook > Activity 1: Setting Up a Highly Available and Secure Development Environment > Development Environment Setup Task Flow > Tuning Unwired Server Performance*.

## **Messages from Shutting Down the Primary Server in a Cluster Environment**

**Problem:** When you shut down the primary server in a cluster environment, you see many error or exception messages in the server log files on secondary nodes.

**Message examples include:** `QueueNotReadyException`, `ConnectFailedException`, `Connection Failed`, and `org.omg.CORBA.COMM_FAILURE` WARN or ERROR

**Explanation:** These errors are caused by switching of the primary server in the cluster, and are expected messages. In general, switching completes in one minute. After one of the secondary nodes becomes the primary node, no such messages are output in log files in the nodes.



## Database Services Do Not Start Automatically

**Problem:** Data services do not start automatically, although previously they did.

**Explanation:** This can happen if the database log path was changed without using **dblog**.

**Solution:** If this is the case, you must supply the correct database log path every time you restart a data service. To avoid having to supply this information repeatedly, use the **dblog** utility to set the correct file name and path. See *Using a Different Database Log Path* in *System Administration* for instructions. You can also see the corresponding database server log file for information (see the *Log File Locations* topic for the log file location).

## Unwired Server Cannot Start in Domain Version Mismatch

**Problem:** The Sybase Unwired Platform Server services in one of the server nodes cannot start if the domain version has a mismatch. In the server log there will be an error message containing "Cluster Synchronization failed. Message : Error retrieving sync data from http URL:"

**Explanation:** This error may be caused by install or uninstalling repeatedly, or receiving an exception during an upgrade..

**Workaround:** manually update the cluster domain version number under  
`<Unwired_Platform_Install_Dir>\UnwiredPlatform\Servers  
 \UnwiredServer\Repository\Instance\com\sybase\sup\server  
 \DomainVersion\domain-version.properties`, making it the same as the other  
 server node.

## Troubleshoot Workflow DCN Failures

**Problem:** An error code and message may be returned to the calling party when a workflow data change notification (DCN) fails.

**Solution:** Resolve the problem and resend the workflow DCN. Unwired Server failures and errors include:

- **Unwired Server down** – HTTP 400 is returned to the calling party
- **Missing DCN parameter, such as cmd=wf** – HTTP 500 is returned to the calling party
- **The workflow DCN message is sent to a non-existent user** – this error message is returned to the calling party:

```
xmlResponse: {"id":"000000392902","success":false,
 "statusMessage":"no workflow message is sent because no
 device is found to send the notification","result":[]}
```

## **No Workflow Message is Sent**

A workflow DCN response is returned if the request is sent to a device that has no users mapped.

```
no workflow message is sent because no device is found
to send the mail
```

Explanation: When a new message-based synchronization Workflow device user is created and registered from Sybase Control Center, by default there are no users mapped to the device (**Device Users > Devices > MBS user name > Users....** In this case, when sending a Workflow DCN request, the response is similar to:

```
{"id":"dcntest_test_001","success":false,"statusMessage":
"no workflow message is sent because no device is found to send the
mail"
, "result":
[{"id":"dcntest_test_data_1","success":true,"statusMessage":""}]}
```

**Workaround:** Invoke any other Unwired Platform application (for example, a client-initiated or server-initiated workflow application, or a native application), which maps an Unwired Platform user to a MBS device user, then send the DCN with

```
"to":" < this SUP user > "
```

# 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>&lt;username&gt;</i> \Application Data\Macromedia\Flash Player\#SharedObjects
Windows Vista	C:\Users\ <i>&lt;username&gt;</i> \AppData\Roaming\Macromedia\Flash Player\#SharedObjects
Macintosh OS X	/Users/ <i>&lt;username&gt;</i> /Library/Preferences/Macromedia/Flash Player/#SharedObjects
Linux	/home/ <i>&lt;username&gt;</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` 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` 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\ScsADataserver\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 server-side configuration by trying to connect to Unwired Server from Unwired Workspace (requires creating Unwired Server Connection Profile with proper user name and password among other things).
- Check `SCC_HOME\log\agent.log` to see if the authentication failed. If so, check the security configuration. By default, Sybase Control Center shares the same configuration as the one used on the local Unwired Server. Ensure that the correct login and password is provided (one that has administrative privilege). See *Enabling Authentication and RBAC for Administration Logins* in the *Security* guide.
- If all services are running, check the `SCC_HOME\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.

### Administrator Account is Locked

Problem: An administrator tried logging into Unwired Server from Sybase Control Center multiple times. After receiving multiple instances of the message Wrong username and password errors, the message The account is currently locked. Please contact your server administrator. is finally displayed.

Explanation: The user has exceeded the threshold for failed login attempts. The platform administrator sets the properties that control the login failure account lock threshold, and the timeout period. When a user exceeds the threshold value, the account is locked for the timeout period.

Solution: A user must wait for the lock timeout value to pass, and then log in again.



For details, see *Creating a Security Configuration* in *Sybase Control Center for Sybase Unwired Platform* 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\Flash 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](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 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

## Troubleshoot the Sybase Control Center

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](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.

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

## Troubleshoot the Sybase Control Center

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:
  - a. 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.
  - b. 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:

- 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:

## Troubleshoot the Sybase Control Center

- 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 there is a conflict on the currently configured port for the administration web service or a component of 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`.

---

**Note:** This link works from the host where Sybase Unwired Platform is installed, using the correct Messaging port. 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.

## Troubleshoot the Sybase Control Center

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

## Troubleshoot the Sybase Control Center

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

## Troubleshoot the Sybase Control Center

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. Start by looking at the Sybase Control Center log file `SCC_HOME\log\gateway.log` for the error message that occurred when user interface displayed the message. In most cases, the error is an unexpected failure condition in the server, and further details can be obtained by reviewing the `SUP_HOME\Servers\UnwiredServer\logs\{ServerName}-server.log`. If that does not help, contact your Sybase Unwired Platform technical support representative.

## **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 `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:

## Troubleshoot the Sybase Control Center

- 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.
    - 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.



# Troubleshoot Mobile Business Object Development

Provides troubleshooting topics for problems you may encounter while using Sybase Unwired Platform to develop, test, and deploy mobile business objects (MBOs).

See the *Validation Rules* section for information about validation rules and error messages.

## Unwired WorkSpace Crashes When Previewing or Test Executing Large Amounts of Data

---

**Problem:** When previewing attributes or test executing operations that involve large amounts of data in Unwired WorkSpace, for example, either a large number of rows, or large object in the returned result, Unwired WorkSpace can run out of memory and crash.

**Solution:** Either do not preview or test execute large results, or increase the java VM arguments `-Xss`, `-Xms`, and `-Xmx` in the `UnwiredWorkSpace.bat` file to increase the stack and memory (heap) size and restart Unwired WorkSpace.

## Cannot Start Unwired WorkSpace After Modifying Plugins

---

**Problem:** After applying private patches, temporary fixes, or manually changing plugin contents, Unwired WorkSpace fails to start.

If Unwired WorkSpace fails to start in any of these scenarios, you may have a corrupt p2 and/or configuration folder under `SUP_HOME\Eclipse`.

**Solution:** The installation program saves clean p2 and configuration folders for recovery purposes. Follow these steps to recover corrupt p2 and configuration folders:

1. Ensure Unwired WorkSpace is shut down.
2. Delete the p2 and configuration folders under `SUP_HOME\Eclipse`.
3. Unzip the clean `Eclipse3.7.2_p2_configuration.zip` file, located in `SUP_HOME\MobileSDKxx\Eclipse\`.
4. Restart Unwired WorkSpace.

## Cannot Access Sample Database After Switching Network Types

---

**Problem:** After switching network types, Unwired WorkSpace is unable to connect to the predefined "My Sample Database" connection profile.

The "My Sample Database" connection profile, by default, uses 'localhost' as the hostname. But if you change it to the real hostname, then when you switch network types, the host may not be recognized. For example, moving a laptop from an office ethernet to an WiFi connection outside the company's firewall may prevent access to the database even if it is on the same machine as Unwired WorkSpace.

**Solution:** In Unwired WorkSpace, modify the connection profile host property to indicate localhost as the Host. To verify the connection is working, click **Test Connection** to ping the database. You may also have to redeploy any projects that access an Unwired Server SQL Anywhere database connection pool and reestablish the Unwired Server connection to the database by modifying the same property from hostname to localhost.

## Data Source Connection Fails After Installing to a New Directory

---

**Problem:** After uninstalling Sybase Unwired Platform, then installing to a new directory, a data source connection fails in Unwired WorkSpace, with the message:

```
java.lang.ClassNotFoundException:
com.sybase.jdbc3.jdbc.SybDriver.
```

**Explanation:** This may happen if you use an old WorkSpace, and an old JDBC connection profile with its `jconn3.jar` pointing to the old directory.

**Solution:**

1. Once Unwired WorkSpace is installed, switch to a new WorkSpace directory.
2. Or, manually change the driver path by modifying the properties of the connection profile for a New Driver Definition.

## JDBC Connection Errors

---

**Problem:** The JDBC connection succeeds but the error log receives continuous errors.

**Explanation:** This can occur if the connection includes any incorrect property which could cause a connection failure (for example, non-existing database name, incorrect URL format (specific to different types of database servers), referencing the wrong JDBC driver, or the JDBC driver does not exist (if you have not downloaded the driver)).

Solution: When you create the connection, ensure that all properties are correct, and that you have the required drivers.

## **Mobile Business Object Errors**

---

Provides information for problems you may encounter while developing mobile business objects.

### **Binding an MBO to a Database Fails**

Problem: Failed to bind a mobile Business object (MBO) to a non-Sybase database due to a missing JDBC driver.

Solution: If you are using a IBM, DB2, Oracle, or Microsoft SQL database, you must place the corresponding JAR file into the `lib` folder in the associated plugin. Before you create or bind an MBO using any of these databases, select the JAR from the appropriate directory when you create the corresponding JDBC. See *Configuring Your Environment to Use a JDBC Driver in Unwired WorkSpace - Mobile Business Object Development*.

### **Multilevel Insert Operation Fails**

Problem: A multilevel insert operation fails on the device client.

Explanation: A multilevel insert operation can fail for a variety of reasons, such as the attribute and parameter relationship are not set up properly, or the relationship definition is not correct.

Solution: Investigate how the multilevel insert operation is designed, and make appropriate corrections. See the *Understanding Multi-level Insert Operations* topic in *Unwired WorkSpace - Mobile Business Object Development*.

### **MBO Retrieves Incorrect Data**

Problem: The MBO does not retrieve the expected data on the device.

Explanation: When you create a relationship to a mobile business object and define a synchronization parameter for the child object, the child object can no longer be used to retrieve all the items. If you want the MBO to retrieve all items, you must create a new MBO without relationships, or create the relationship to the MBO and set the synchronization parameter. Set the synchronization parameter with a default or a personalization key, or allow the user to input the value of the synchronization parameter.

Solution:

1. To retrieve all items, you must create a new MBO.
2. To use the original child mobile business object without a relationship, set the synchronization value on the device. For example, if you create a screen with only the child mobile business object, then you must set the synchronization parameter with

personalization. If you do not set personalization, then no data is shown on the device for the client MBO.

### **Relationship Error**

**Problem:** Error occurs when trying to link multiple mobile business objects (MBOs) to a single child MBO.

You see an error similar to this in Unwired WorkSpace:

```
Error: The target mobile business object 'MBO_name' can not be used in relationship 'relationship_name' since it already was used in a one-to-many or one-to-one relationship as a target.
```

When a child mobile business object (MBO) has multiple parent MBOs, only one composite relationship is allowed.

**Solution:** Examine all of the relationships to a single child MBO and unselect **Composite** for all but one.

---

**Note:** Relationship errors can happen for a variety of reasons. For information about relationship validation rules and development guidelines, see:

- *Troubleshooting > Validation Rules > Mobile Business Object Validation Rules and Error Messages > Relationship Validation Rules and Error Messages*
  - *Sybase Unwired WorkSpace - Mobile Business Object Development > Develop > Working with Mobile Business Objects > Mobile Business Object Data Properties > Creating Relationships Between Mobile Business Objects > Relationship Guidelines*
- 

### **Warning Message When Creating Relationships**

**Problem:** When attempting to create a relationship between different types of MBOs, a warning message is displayed.

**Explanation:** A relationship between a normal MBO and a structure is not allowed; and a relationship between a normal MBO and a local business object is not allowed. Relationships can be created only between normal MBOs, or between local business objects. No cross-over cases are supported.

**Solution:** Create a relationship between normal MBOs, or between local business objects.

### **Troubleshooting Default Values for Adaptive Server® Anywhere uniqueidentifiers**

Keep these points in mind when working with Adaptive Server Anywhere uniqueidentifiers.

1. Uniqueidentifiers are treated as binary datatypes in Unwired WorkSpace. If the enterprise information system (EIS) on which the uniqueidentifier resides:

- Accepts NULL as a valid value, you can set the default value to NULL.
  - Does not accept NULL as a valid value, you need to set the correct default value. By default Unwired WorkSpace uses an empty string as the default value for binary datatypes.
2. To avoid entering values for the UNIQUEIDENTIFIER columns during insert operations, modify the default SQL definition as this example illustrates:
    - a. Create an Adaptive Server Anywhere table with SQL:

```
CREATE TABLE dba.YorkTable (
 UniqueColumn UNIQUEIDENTIFIER NOT NULL DEFAULT NEWID(),
 Characters VARCHAR(10) NOT NULL,
)
IN SYSTEM
;
ALTER TABLE dba.YorkTable
ADD CONSTRAINT ASA126 PRIMARY KEY NONCLUSTERED (UniqueColumn)
;
```

- b. Drag-and-drop the table to create the new MBO.

The auto-generated Create operation SQL is:

```
"INSERT INTO sampledb.dba.YorkTable (UniqueColumn, Characters)
VALUES
(:UniqueColumn, :Characters)"
```

- c. To avoid entering values for the UNIQUEIDENTIFIER column during insert operations, change the SQL definition to:

```
"INSERT INTO sampledb.dba.YorkTable (Characters) VALUES
(:Characters)"
```

3. For operations that use uniqueidentifier datatype argument default values, change the default operation argument datatype to a String instead of the automatically generated Binary from the Operation Parameters tab in Unwired WorkSpace.

## AutoCommit Option for JDBC MBOs Using an ASE Data Source

When you create MBOs from Adaptive Server® Enterprise stored procedures that use temporary tables, you must select the **AutoCommit** check box in the New Attributes or New Operation definition screen.

An error message displays in Unwired WorkSpace if Auto Commit is not selected. For example, create a temporary table named “tempstores” in this stored procedure:

```
CREATE PROCEDURE dbo.ase_sp
AS
BEGIN
 create table tempstores (temp_row_id integer, temp_id integer)
 insert tempstores select 1, 1
 insert tempstores select 2, 2
 insert tempstores select 3, 3
 select temp_row_id, temp_id from tempstores group by temp_row_id
 drop table tempstores
END
```

If this stored procedure is used to model an MBO (attributes and operations), when you preview an operation, this error message displays:

```
The 'CREATE TABLE' command is not allowed within a multi-statement transaction in the 'tempdb' database.
```

### Passing an Empty String or Byte into a Required Binary Column in ASE

When defining an MBO from an Adaptive Server Enterprise data source an error message displays if you pass an empty string or empty byte[] into a required binary column in ASE results in an ASE non-nullable column exception.

1. Create the MBO. For example, drag and drop an ASE data source onto the mobile application diagram.
2. Create a Create operation named create1.
3. In the Parameters page of the New operation wizard, click **Next** and it throws a `RuntimeException` if performing the described action. For example:

```
Error executing on back-end server:
java.lang.RuntimeException: java.lang.RuntimeException:
com.sybase.jdbc3.jdbc.SybSQLException: The column binary1
in table pubs3.dbo.AllDataTypes does not allow null values.
```

4. Running 'Test Execute' for the Create operation also throws an error:

```
Execution failed: The column binary1 in table
pubs3.dbo.AllDataTypes does not allow null values
```

**Solution:** use "IA==" as the default value for the binary(1) argument. Create a new Create operation, and select 'Test Execute' to verify it works correctly.

### Preview Errors for a Web Service Mobile Business Object with date/time Types

**Problem:** You see an error when you are previewing a Web service mobile business object (MBO).

**Explanation:** If the Web service **datetime** datatype includes time zone information, the preview data may not be correct. This issue stems from the different time zones between the enterprise information system (EIS) and the device client.

**Solution:** See *Unwired WorkSpace - Mobile Business Object Development* for information about time zone datatype behavior, and modify the Web service MBO accordingly.



## **Web Service Mobile Business Object Must Be Connected to Connection Profile**

Problem: The Web service connection profile must be connected to deploy, generate code, build packages, and so on.

Explanation: If you try to perform any of the above tasks without being connected to the Web service connection profile, a null `OperationMetaData` exception error message displays and the task fails.

Solution: Ensure `UnwiredWorkSpace` is connected to the connection profile to which the Web service MBO is bound.

## **Refresh Fails Due to Missing Load Argument Default Values**

Problem: Adding load arguments with "Propagate to attributes" that have no corresponding datasource columns, then selecting Refresh, may fail due to missing load argument default values.

Unmap the "Propagate to attribute" field, which allows you to specify the necessary default values so that the execution to the EIS successfully retrieves MBO metadata.

## **Input Parameters Fail to Generate for Complex Data Sources**

Problem: For Web service and other data sources with very complex structures with deeply nested element trees, `StackOverflow` or `OutOfMemory` may occur, or some of the operation input parameters may not be generated properly.

Explanation: When a WSDL or other complex data source is wsdl is deeply nested, long processing times and memory errors can result.

Solution: Increase the stack and heap size of the Eclipse runtime parameter in `UnwiredWorkSpace.bat` with `-Xmx`, `-Xms`, `-Xss` arguments, or reduce the complexity or nested level of the Web service definition of the MBO operation.

## **Output Mapping Warning Message for Multilevel Insert Operations**

Problem: A warning in the output mapping tab indicates that some output mappings are ignored when using the "Apply Output of ENTITY READ" operation cache policy.

During deployment, output mappings are automatically generated for Create operations to support multi-level insert operations. The set of generated output mappings may not be the same as the output mappings configured in `UnwiredWorkSpace` for the operation. However, the final set of output mappings is a union of the two sets, so the user-configured output mappings are retained. The MBO developer may see a warning in the output mapping tab indicating that some output mappings are ignored when using the "Apply Output of ENTITY READ" operation cache policy.

Ignore the warning since the output mappings are automatically generated during deployment.

### **Attribute Length Set Incorrectly for Operation and Load Arguments**

Problem: Due to JDBC driver metadata retrieval related API limitations, Unwired WorkSpace does not always set the correct attribute length, especially for operation arguments and load arguments. For STRING and BINARY types, if the lengths are unknown, Unwired WorkSpace sets them to STRING(300) and BINARY(32767) by default, and gives a warning to adjust the length to avoid wasting device database space, and/or getting runtime data truncation.

Solution: Examine the EIS database metadata and adjust the attribute length, the types and lengths of operation arguments, and the load arguments in Unwired WorkSpace to avoid runtime errors.

### **Unable to properly update the CDB because the partition was improperly assigned**

Problem: An MBO with a load/synchronization parameter, using an on-demand cache group interval greater than 0, and the load parameter does not have the "Propagate to Attribute" set:

1. A record is created by the device and synchronized to the EIS.
2. The record is deleted in the EIS for any reason.
3. When the on-demand cache group interval expires, the next synchronization request from the device should refresh the CDB and device correctly. Although the newly created record has been deleted from the EIS, it still exists in CDB and device where they remain.

Solution: When designing the MBO, map its load parameter to a corresponding attribute with "Propagate to Attribute". If there is no corresponding attribute, add one, then map it. For example, the Customer MBO has this definition:

```
SELECT id,
 fname,
 lname,
 address,
 city,
 state,
 zip,
 phone,
 company_name FROM sampledb.dba.customer
WHERE state = :state
```

Map the Customer MBO's state load argument to the state "Propagate to Attribute" from the Load Arguments tab.

## **Syntax Errors When Creating a Database Mobile Business Object**

Problem: Calls from Unwired WorkSpace to the EIS during MBO creation/development cannot distinguish syntax from data errors.

When adding an operation and corresponding SQL statement, Unwired WorkSpace executes the SQL statement to verify it is correct. Sometimes the return error is not a syntax error but a data error (for example, a foreign key restriction). The EIS API cannot distinguish syntax from data errors and exits the wizard.

Be aware of the EIS schema to which you are binding the MBO. For example, verify that database data is correct, and enter the correct default values so the SQL statements execute correctly.

## **Sybase MBOs That Invalidate the Cache**

Problem: MBO operations bound to Sybase BAPIs that use the "Invalidate the cache" policy may not refresh the Unwired Server cache.

Since SAP BAPI operations run asynchronously within the Sybase system, the operation may not complete before Sybase Unwired Platform attempts to refresh the cache with new or updated results if you are using the "Invalidate the cache" policy.

Do not use the "Invalidate the cache" policy with MBO operations that are bound to an Sybase BAPI.

## **Deployment Errors**

---

Provides information for problems you may encounter while deploying mobile business object projects.

### **Deploying to Unwired Server Fails**

Problem: Deployment fails with when you deploy a mobile business object (MBO) to Unwired Server.

Explanation: This may happen for a variety of reasons, for example, if the data sources bound to MBOs are not up and running, the Deployment wizard will fail.

Solution: Make sure the data sources associated with the MBOs are up and running.

## **Deploying to Unwired Server in a Cluster Fails**

**Problem:** Deployment fails when you deploy a mobile business object (MBO) to Unwired Server in a cluster.

**Explanation:** When you deploy a package into a Sybase Unwired Platform cluster, you must be connected to the Unwired Server designated as the primary server in the cluster. If not, deployment fails and you see a message similar to following in the server logs:

```
YYYY-MM-DD HH:MM:SS.SSS WARN MMS Thread-466
[com.sybase.sup.server.SUPDomain] On Non primary servers, you could
not
execute createDsyncDataZip()
java.lang.Exception: Current server is not primary, request
rejected.
```

**Solution:** Set up a connection profile to the server that has become the primary, and deploy using that connection. See *Creating a Unwired Server Connection Profile* in *Unwired WorkSpace - Mobile Business Object Development* for how to create a new Unwired Server connection profile.

## **Deploying a Renamed MBO Fails**

**Problem:** Deployment fails with unknown issue error when an MBO that contains custom object queries is renamed.

**Solution:** Modify the object query definition to reference the new MBO name.

## **Troubleshoot Code Generation**

---

Provides troubleshooting topics for generating device application code using the client object API.

### **Code Generation Fails**

**Problem:** Code generation fails or takes a very long time.

**Explanation:** This may indicate that the data sources bound to mobile business objects (MBOs) are not up and running, which prevents the binding to take place. The resulting performance impact causes code generation to fail.

**Solution:** Make sure the data sources associated with the MBOs are up and running.

## SQL\_MAX\_ROW\_SIZE\_EXCEEDED

Problem: You encounter a `SQL_MAX_ROW_SIZE_EXCEEDED` error during client object API code generation. This means the validation process has detected one or more size problems in your mobile business object (MBO).

Solution: Review your MBO design, and make the appropriate modifications using this information. Some possibilities:

- Set the maximum length of string attributes to a smaller number (the default value is 300).
- Change the maximum length of string attributes to 8192. Internally, the SQL datatype of this attribute will be converted to "LONG VARCHAR".
- Redesign the MBO by removing some attributes.
- Use the proposed page size in the code generation wizard.

## Namespace Name Does Not Exist Error

Problem: When generating C# code, this error appears: The type or namespace name `<'name'>` does not exist in the namespace `'SUP.name.com'` (are you missing an assembly reference?).

The full message, including an example, appears as:

```
public sealed class SalesforceAutomationDB :
com.sybase.afx.DatabaseBase
{
....
}

Error 1 The type or namespace name 'sybase' does not exist in
the namespace 'SUP.Mobile.com' (are you missing an assembly
reference?)
C:\Documents and Settings\username\My Documents\Visual Studio
2008\Projects\SFA\SFA\Generated Code\src\SUP\Mobile\com
\SalesForceAutomationDB.cs 41 54 SFA
```

Explanation: The namespace name does not follow C# namespace conventions. In this example, avoid using patterns that include `com`, and embedding a name, such as `Sybase`:

- `(xxx.)+com(.xxx)*`
- `(xxx.)+Sybase(.xxx)*`

Solution: Change the namespace name to follow C# namespace conventions, and regenerate C# code. In this example, the conventions would not use `com`, and would include the `Sybase` name in the format `Sybase.Persistence`.

## Cannot Generate Code with Javadoc Option Enabled

Problem: Code generation fails when trying to generate code to a file system folder named with Chinese characters with the **generate javadoc** option selected.

Explanation: This typically occurs if the folder name contains Chinese characters.

Solution: To generate the code to a folder containing Chinese characters, you must first change the Language for non-unicode programs to Chinese.

1. From the Start menu, select **Settings > Control Panel**.
2. Double-click Regional and Language Options, then click the Advanced tab.
3. From the drop-down list, select Chinese, and click Apply.
4. Click OK.

## Installing the net\_rim\_api.jar or android.jar File

Depending on your Java development environment, install either the `net_rim_api.jar` file (BlackBerry) or the `android.jar` file (Android) to avoid code generation errors when generating Java code for native application development.

After generating Java code for BlackBerry or Android, error icons appear next to the project for which you generated the code. These errors can be viewed in the Problems view and usually in the project's `<MobileApplicationProjectName>DB.java` file, and appear because of dependencies on the `net_rim_api.jar` or `android.jar` file if not in the project build path. You can avoid these errors by following this procedure for any projects for which you generate Java code:

1. Install the BlackBerry or Android development environment.
2. During code generation, set **Third-party jar file** to the appropriate JAR file. For example, for BlackBerry locate `net_rim_api.jar` file, which depends on where the JDE is installed. For example, `C:\Program Files\Research In Motion\BlackBerry JDE 5.0.0\lib`.

## Error Messages About Missing Directories When Generating Object API Code

Problem: When the MBO developer regenerates ObjectAPI code from the Unwired WorkSpace Code Generation wizard, if **Clean up destination before code generation** is selected in the Configure Options page, an error similar to this appears in the Error Log view:

```
Resource '/<Mobile Application Project name>
/Generated Code/temp_afx does not exist. Resource '
/<Mobile Application Project name>/Generated Code/genfiles'
does not exist.
```

Ignore the error messages. The objectAPI code generates successfully.

## **Troubleshoot Unwired WorkSpace Errors**

---

Provides troubleshooting topics for Unwired WorkSpace errors.

### **Unwired WorkSpace Does Not Support Saving Files as PDF**

Problem: Invoking **File > Save As Image File** from the Object Diagram context menu fails if PDF is the image format.

This is a known Eclipse GMF bug. See [https://bugs.eclipse.org/bugs/show\\_bug.cgi?id=338380](https://bugs.eclipse.org/bugs/show_bug.cgi?id=338380)

### **Installing Unwired WorkSpace Into a Different Directory**

Problem: If you uninstall Unwired WorkSpace, then reinstall it in a different directory, you get an `ClassNotFoundException` message if the old workspace is used and you connect using the JDBC connection profile.

To avoid this error, update the driver definition to point to the correct path:

1. From Unwired WorkSpace, select **Window > Preferences > name > Connectivity > Driver Definitions**.
2. Select the driver type, and click **Edit**.
3. Click the **Jar List** tab, and edit the path of the JDBC driver.
4. Reimport any mobile application projects that require the updated driver for connecting to the database.

### **Avoiding Out of Memory Errors During Preview and Test Execute**

Problem: During MBO development, if the data volume is large, for example, either a large number of rows, or a large object in the returned result, Unwired WorkSpace may run out of memory and fail when you click **Preview** or **Test Execute**.

To avoid these errors, either do not select the Preview or Test Execute options, or increase the Java VM arguments `-Xss -Xms`, and `-Xmx` in `UnwiredWorkSpace.bat` to increase the stack and memory (heap) size.

### **OSGI Bundle Related Error Messages**

Problem: After starting Unwired WorkSpace, you may see Eclipse OSGI bundle related error message(s).

These errors appear in the Error Log view, for example, `org.eclipse.osgi.framework.internal.core.AbstractBundle$BundleStatusException`.

You can safely ignore these errors.

## **Unwired WorkSpace Crashes and Fails to Restart**

Problem: After Unwired WorkSpace crashes, the current Eclipse workspace retains inconsistent states that may prevent a successful restart.

This is a known Eclipse issue. Remove the following file from the problematic Eclipse workspace, then restart Unwired WorkSpace: <The workspace used in the previous crash>\.metadata\.plugins\org.eclipse.core.resources\n.snap.



# Troubleshoot Object API Development

Provides troubleshooting topics for developing device applications using the client object API.

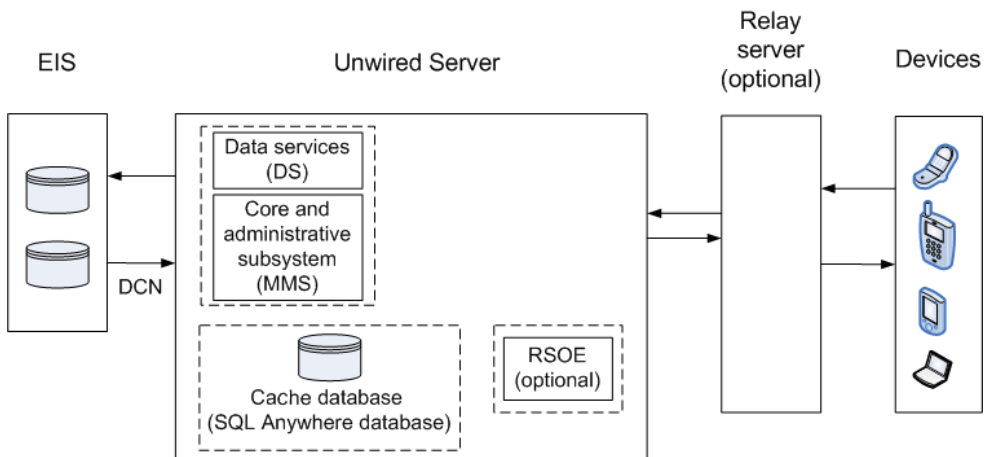
## Debugging Replication Synchronization

Describes the replication synchronization process flow, and options available for diagnosing problems at each step of the process.

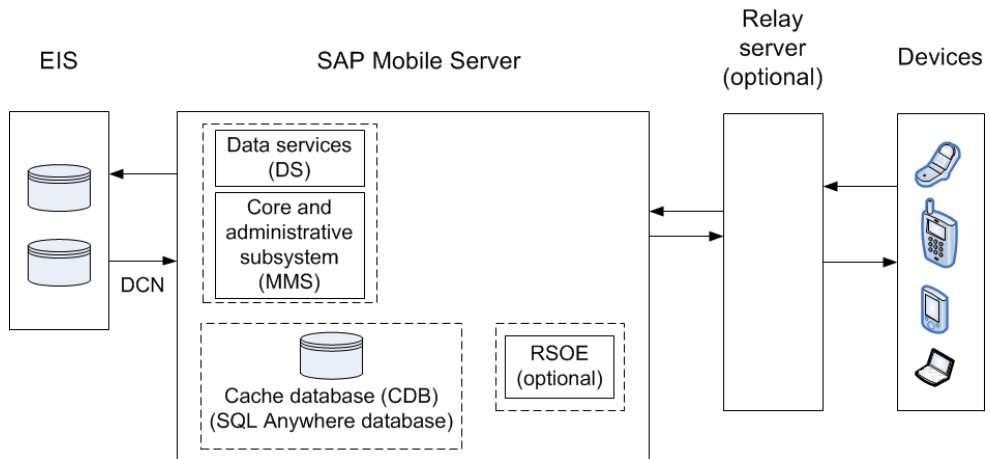
### Process flow

This diagram shows a logical view of replication synchronization communications between the enterprise information system (EIS), Sybase Unwired Platform components, and device clients. Relay server is included as an optional component.

**Figure 13: Replication synchronization process flow**



**Figure 14: Replication synchronization process flow**



*Troubleshooting options*

This section describes the options available for diagnosing replication synchronization processing flow problems.

Unwired Server – data services (DS):

- In Sybase Control Center, enable data services (DS) subsystem logging by setting its log level. The output is written in the server log.
- Review subsystem messages for information about getting data and sending transactions.
- For cache activities, you can enable domain logging on the package using Sybase Control Center, and then review data in cache sub-systems (Cache Refresh and DS Interface specifically) for the period domain logging is enabled. See *Domain Logs* and *Domain Log Categories* related documentation in *System Administration* and *Sybase Control Center for Sybase Unwired Platform* online help for complete details.

Unwired Server – core and administrative subsystem (MMS):

- If you like to troubleshoot at the system-level (which will record all domains and packages activities), you can enable MMS logging by setting the log level in Sybase Control Center. The output is written in the server log.
- Select the desired application package, and enable domain logging on that package. Review the captured log messages in Sybase Control Center.
- For quick troubleshooting to locate system errors, you can check the package-level client log and MBO/Operation history in the Sybase Control Center.
- View the MBO/Operation history in Sybase Control Center for errors during back-end interaction with the EIS.
- On the data-tier node, review the database error logs:

- `errorlog.txt` – database system errors
- `clusterdb_errorlog.txt` – cluster DB server log
- `monitordb_errorlog.txt` – monitor and domainlog DB server log

Relay server – if used:

- Enable relay server tracing by setting verbosity to 1 or higher. You can do that by editing the `rs.config` file of your Relay Server and running the `rshost` utility. Check the *System Administration* topic on Relay Server Setup.
- Review the relay server log files. The logging for `rshost.exe` is directed to the `myrshost.log` file in the `C:\temp` directory.
- Check the IIS log files (`windows\logfiles\W3SVC1` and `HTTPERR`). The IIS server log path is configurable from `inetmgr` from **Default Web Site > Properties > Website tab > Enable Logging > Properties > Log file directory**. The recommended location is `C:\Inetpub\LogFiles`.

On Unwired Server, check the Sybase Unwired Platform outbound enabler log for replication synchronization entries: `SUP_HOME\Servers\UnwiredServer\Logs\NodeName.RSOEnum.log` (for example, `smpqa01SMPServer1a.RSOE1.log`).

Device clients:

- Use the platform-specific client logging API (**com.sybase.persistence.LogRecord** in Java, and **Sybase.Persistence.ILogRecord** in C#) to capture log messages and upload them (Client Log) to Unwired Server. You can view these messages in the Sybase Control Center user interface

## Insert Operation Fails for iOS

---

**Problem:** A mobile business object (MBO) insert operation fails with a runtime exception for iOS; however, the same MBO works on Windows Mobile.

**Solution:** For an attribute that cannot be null, assign a value if the corresponding Property datatype in the generated MBO Objective-C code is a class (pointer).

When developing an iOS client:

- If a non-nullable attribute datatype is converted to a nonprimitive datatype (for example, class `NSNumber` or `NSDate`) in the iOS client application, verify that the corresponding property for the MBO instance is assigned a value; otherwise the application may issue a runtime exception when creating a new MBO instance.
- If an Adaptive Server Enterprise column datatype is numeric, during the MBO creation, the datatype of the corresponding attribute is decimal and non-nullable. In generated Objective-C MBO code, the corresponding datatype is `NSNumber`. When creating a new

MBO instance, assign this property a value, otherwise the application may issue a runtime exception (its actual value is later replaced by Unwired Server).

### MBO Update Fails

---

**Problem:** A mobile business object (MBO) that you edited using the C# client API fails to update upon synchronization.

**Solution:**

1. Check the Logs Screen for the detailed log record.
2. If the log record does not provide enough information, check the C# client API code for errors, for example:

```
int id = 777; //it's the customer id
Customer cust = Customer.FindBy(id);
cust.Name = "some other name";
cust.Update();
cust.SubmitPending();
DBClass.Synchronize();
//...
//you verify operation log that indicates the update failed at
backend server
//...
cust = Customer.FindBy(id);
cust.Delete();
cust.SubmitPending();
DBClass.Synchronize();
//...
//you verify operation log that indicates the delete failed
//...
```

**Call `CancelPending()`** after the failed update operation to delete the MBO, for example:

```
int id = 777; //it's the customer id
Customer cust = Customer.FindBy(id);
cust.Name = "some other name";
cust.Update();
cust.SubmitPending();
DBClass.Synchronize();
//...
//you verify operation log that indicates the update failed at
backend
servercust = Customer.FindBy(id);
//...
Customer customer = Customer.FindById(id);
if(errorHappened)
{
 Customer.CancelPending();
}
else
{
 customer.SubmitPending();
}
```

```
//...
//you verify operation log that indicates the delete succeeded
//...
```

## Windows Mobile Compile Error For Generated API Project

Problem: A Windows Mobile compile error is encountered when generating a client object API project.

For example: Error 6 'entity.<variable>' does not contain a definition for '<variable>' and no extension method '<variable>' accepting a first argument of type 'entity.<variable>' could be found (are you missing a using directive or an assembly reference?) C:\work\entity\_Sol\entity\_MBO\entity\intrnl\Department\_rs.cs 45 46 entity\_MBO

Explanation: This error is due to variable name conflicts. The package name includes client object API reserved words (in the example, "User" is used by OfflineAuthentication.cs, and "entity" is the field of an internal operation). The compile process could not resolve package name with the internally reserved words, and reported the error.

Solution: Rename the package, then rerun the compile.

## Cannot Insert DEFAULT in Timestamp Column

Problem: Cannot insert DEFAULT in the timestamp column using the Object API.

Explanation: DEFAULT is a Microsoft SQL keyword, and cannot be set by the client.

Solution: Use this create operation instead:

```
insert into sql_date (c_datetime, c_smalltdt) values (:dt1, :dt2)
```

## Synchronization to SAP Data Source Repeats Excessively

Problem: Created an MBO that is calling a customized RFC with an input parameter. The first synchronization to the SAP data source works correctly. The second sync executes the first RFC, then a second one. This causes failures in SAP, because the new update cannot be transmitted.

### *Client Names*

Explanation 1: Synchronization parameter crosstalk between clients could be causing a problem if multiple clients are running concurrently using the same name. Synchronization parameters are stored on Unwired Server by user name.

Solution: Avoid the issue by running each client with a different user name. It is also necessary to pass the username/password combination using `LoginToSync`.

### *MBO Read Operations*

Explanation 2: If you map your MBO's read operation (through which you retrieve the MBO attributes) to a backend operation that commits a BAPI, then keep in mind that each time you perform a synchronization, the sync parameter is stored on Unwired Server. Later, when you sync it again with another sync parameter value, all the stored sync parameter values are synced again.

Solution: To avoid this problem, call `SyncParameter.delete()` to clear the cache after each synchronization, for example:

```
<name>SynchronizationParameter param =
nameMBO.getSynchronizationParameter();
param.delete();
param = nameDB.getSynchronizationParameter(); //must re-get the sync
parameter instance
param.set... //set new sync parameter value
param.save();
```

## Error When Running Generated 32-bit Windows Code on a 64-bit Machine

---

Problem: An error occurs in the client API code when myDatabase initializes.

Solution: If you encounter an error when running generated 32-bit Windows device client code on a 64-bit machine:

1. In Visual Studio, in the toolbar, select **Configuration Manager** from the Solution Platforms list.
2. In the Configuration Manager, in Active Solution Platform, select **New**.
3. In **Type or select the new platform**, enter x86.
4. Restart the application.

## Error with Multiple Writer Threads Modifying the Same Table Row

---

Problem: When an application enables multiple writer threads, there may be an exception thrown when the multiple writer threads try to modify the same row of a table.

Explanation: This occurs on the KeyGenerator since most of the create, update, or delete methods on the MBO need to use the KeyGenerator to generate new keys. However, multiple threads can write to different Local Business Objects.

Solution: Design application to avoid having multiple threads write to the same table row.

## Relaunched Application Gets Encryption Key Error

---

Problem: When an application is relaunched, it gets the following error message:

```
com.sybase.persistence.PersistenceException: UltraLiteJ
Error[-840]: Incorrect or missing encryption key
```

Explanation: There are messages queued in the messaging client, and the message handler starts to process messages before encryption is set.

Solution: Make sure `Application.getInstance().startConnection()` is called after encryption key is set.

## Encryption Key Cannot Contain Special Characters

---

Problem: When you set and attempt to use an encryption key, you get an `SUPClientException` and synchronization fails.

Explanation: The encryption key does not support special characters.

Solution: Avoid using any of these special characters when you set the encryption key: " ~ ! @ # \$ % ^ & \* ( ) \_ + = - ; : . ' < > / ?

## MinCheckAffectedRowCount behavior with JDBC endpoints

---

Understand how `MinCheckAffectedRowCount` behaves with JDBC endpoints.

For JDBC endpoints only, Sybase Unwired Platform tests all UPDATE and DELETE operations for the number of rows affected in the JDBC backend (This does not apply for JDBC UPDATE/DELETE operations modeled with a stored procedure). If more than one row was affected for an UPDATE or DELETE operation, Sybase Unwired Platform creates an error log entry for the client that simply indicates "com.sybase.vader.da.jdbc.MinCheckAffectedRowCountException" has been encountered on Unwired Server. The cause of the exception is application specific and could mean several things:

1. The predicate definition of the UPDATE or DELETE operation is invalid.
2. The backend RDBMS no longer contains a row that matches the predicate of the UPDATE or DELETE operation.

3. The client application passes in an invalid parameter to the UPDATE or DELETE operation and therefore the predicate of the UPDATE or DELETE operation is failing.

For example, a client deletes a record and synchronizes, and a second client attempts to update the deleted record and synchronizes. The second client correctly receives a replay failure with an error log entry "com.sybase.vader.da.jdbc.MinCheckAffectedRowCountException:null", and the information returned when the `onReplayFailure` callback method is invoked for the entity is empty. Since only the "surrogate" key is available and this fails to find a record, any failure message itself is meaningless.

For another example, see Hybrid App development topic *MincheckAffectedRowCountException Error Received*.

## Problems with a Slow Query

---

**Problem:** A query (an object query, or a `findWithQuery` or `executeQuery`) is slow.

**Explanation:** Some attributes of the MBO used in the query's `where` clause or `join` conditions are not indexed.

**Solution:** Try to add necessary indexes for those attributes used in `where` clause or `join` conditions.

You can only add indexes when defining object queries. To add indexes by defining object queries, in *Unwired WorkSpace - Mobile Business Object Development* see *Creating Object Query Indexes*.



# Troubleshoot Hybrid Web Container Development

Provides troubleshooting information for Hybrid Web Container and Hybrid App development, and Hybrid App package generation.

See the *Validation Rules* section for information about validation rules and error messages.

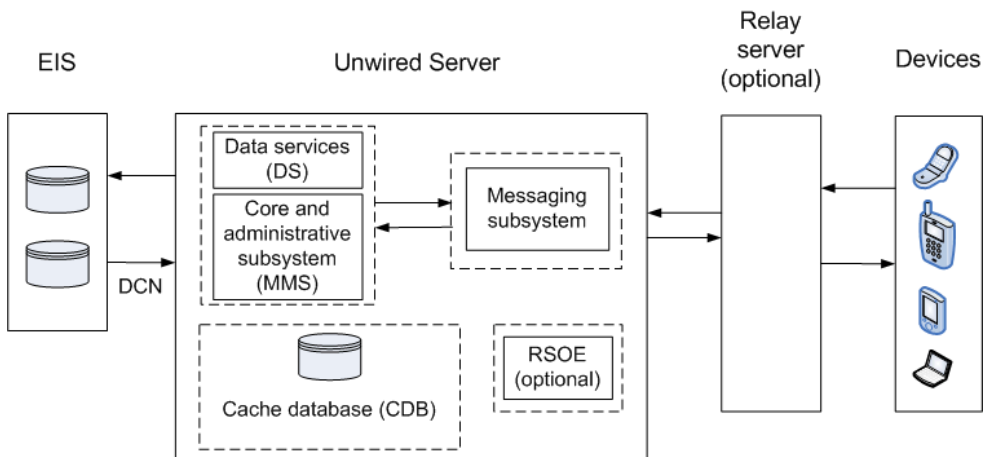
## Debugging Message Synchronization

Describes the message synchronization process flow, and options available for diagnosing problems at each step of the process.

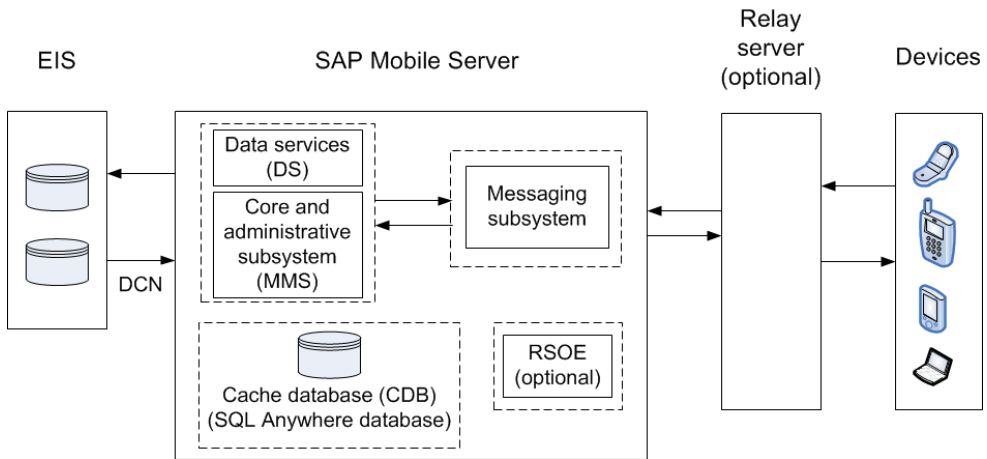
### Process flow

This diagram shows a logical view of message-based synchronization communications between the enterprise information system (EIS), Sybase Unwired Platform components, and device clients. Relay server is included as an optional component.

**Figure 15: Message synchronization process flow**



**Figure 16: Message synchronization process flow**



### *Troubleshooting options*

This section describes the options available for diagnosing message synchronization process flow problems.

Unwired Server – data services (DS):

- In Sybase Control Center, enable data services (DS) subsystem logging by setting its log level. The output is written in the server log.
- Review subsystem messages for information about getting data and sending transactions.
- For cache activities, you can enable domain logging on the package using Sybase Control Center, and then review data in cache sub-systems (Cache Refresh and DS Interface specifically) for the period domain logging is enabled. See *Domain Logs* and *Domain Log Categories* related documentation in *System Administration* and *Sybase Control Center for Sybase Unwired Platform* online help for complete details.

Unwired Server – core and administrative subsystem (MMS):

- If you like to troubleshoot at the system-level (which will record all domains and packages activities), you can enable MMS logging by setting the log level in Sybase Control Center. The output is written in the server log.
- Select the desired application package, and enable domain logging on that package. Review the captured log messages in Sybase Control Center.
- For quick troubleshooting to locate system errors, you can check the package-level client log and MBO/Operation history in the Sybase Control Center.
- View the MBO/Operation history in Sybase Control Center for errors during back-end interaction with the EIS.
- On the data-tier node, review the database error logs:
  - `errorlog.txt` – database system errors

- `clusterdb_errorlog.txt` – cluster DB server log
- `monitordb_errorlog.txt` – monitor and domainlog DB server log

Unwired Server – messaging subsystem :

- Use Sybase Control Center to enable server-side logging for <MSG> log subcomponents.
- Retrieve client-side message-based synchronization trace by sending a request to the MBS device using Sybase Control Center (on the **Application Connections** tab, use **Get trace**). A client's log level can be set in the device settings of the Sybase Control Center.
- Configure messaging server-side module level tracing (for users with advanced knowledge).
  - Use Sybase Control Center to set the log levels of individual modules. Select **Servers > <server\_name> > Log > Messaging Server > Settings > <module> > Properties**.
  - If debugging a Hybrid App, turn on WorkflowClient tracing.
  - Check the module logs – SUPBridge and JMSBridge modules, or the WorkflowClient module (to debug Hybrid Apps).
- On Unwired Server, review the `errorlog.txt` for any cache database system errors: `SUP_HOME\Servers\UnwiredServer\logs\errorlog.txt`.

Relay server – if used:

- Enable relay server tracing by setting verbosity to 1 or higher. You can do that by editing the `rs.config` file of your Relay Server and running the `rshost` utility. Check the *System Administration* topic on Relay Server Setup.
- Review the relay server log files. The logging for `rshost.exe` is directed to the `myrshost.log` file in the `C:\temp` directory.
- Check the IIS log files (`windows\logfiles\W3SVC1` and `HTTPERR`). The IIS server log path is configurable from `inetmgr` from **Default Web Site > Properties > Website tab > Enable Logging > Properties > Log file directory**. The recommended location is `C:\Inetpub\LogFiles`.

On Unwired Server, check the Sybase Unwired Platform outbound enabler log for replication synchronization entries: `SUP_HOME\Servers\UnwiredServer\Logs\NodeName.RSOEnum.log` (for example, `smpqa01SMPServer1a.RSOE1.log`).

Device clients:

- For Hybrid Apps running on Android, BlackBerry, and Windows Mobile, run the **Settings** application, and choose **Show Log**. The client system log for messaging activities appears. Key information includes messages related to server connectivity, server messages, and messaging client requests. Native applications may or may not include **Settings** and **Show Log**, depending on the design.
- Set the device log trace level on the device and get the device logs from Sybase Control Center by selecting **Applications > Application Connections > <device> > Properties > Device Advanced >** and setting the **Debug Trace Level** to 4 or higher.

After running the application on a device, select the device in Sybase Control Center and select **Get Trace**. The client trace log on the device is transferred to the Unwired Server at `SUP_HOME\Servers\UnwiredServer\logs\ClientTrace\`.

## Debugging Hybrid Web Container Development

---

Device client and Unwired Server troubleshooting tools for diagnosing Hybrid Web Container development problems.

### *Client-side debugging*

To find out more information on the device client side:

- Hybrid Web Client (located on the mobile device; not available on iOS) – shows the interaction between the Hybrid Web Client on the mobile device and Unwired Server. On a mobile device that has the Hybrid Web Client installed, view **Settings > Show Log**.
- Unwired Server – shows activity occurring on Unwired Server. Change the tracing level for Unwired Server from Sybase Control Center. In the left pane, click **localhost@localhostnode > Configuration**; and in the right pane click **Log Setting > > Unwired Server** and its modules to set server log level. See *Configuring Unwired Server Log Settings* in *Sybase Control Center for Sybase Unwired Platform*.

### *Server-side debugging*

To find out more information on the Unwired Server side:

- WorkflowClient – shows the interaction between the Hybrid App and the data sent to and from the Unwired Server (to enable, turn Hybrid App tracing on). Change the tracing level for Hybrid App packages from Sybase Control Center. In the left pane click **localhost@localhostnode > Configuration**, and in the right pane click **Log Setting > MessagingServer > WorkflowClient > Properties**, then select the debug Level.
  - For system administrators, see *Configuring Messaging Server Log Settings* in *Sybase Control Center for Sybase Unwired Platform*.
  - For developers, see *Debugging Custom Code* and *Configuring Mobile Workflow Tracing in SCC* in *Developer Guide: Hybrid Apps*.
- Unwired Server – shows activity occurring on Unwired Server. Change the tracing level for Unwired Server from Sybase Control Center. In the left pane, click **localhost@localhostnode > Configuration**; and in the right pane click **Log Setting > Unwired Server**, and its modules to set server log level. See *Configuring Unwired Server Log Settings* in *Sybase Control Center for Sybase Unwired Platform*.

## Debugging Hybrid Web Container Connection Issues

---

List of things to check if the Hybrid Web Container is not connecting to Unwired Server.

*Check the Hybrid Web Container connection settings configuration on the device*

- Verify that there is a valid Unwired Server name in the configuration.  
If the Unwired Server name has changed, you must uninstall and reinstall the Hybrid Web Container on the device.
- Verify that the configuration has a port on which Unwired Server is configured to listen for messages. The default port is 5001 but may be set differently by the administrator.
- Verify that the port is listening. Check with the Unwired Server administrator.
- Verify that the user name is in the configuration. If not, enter the user name in the configuration.
- When connecting to Relay Server, verify that the connection ID (CID) is appended to the URL suffix, for example, /rs/client/rs\_client.dll/<CID>/tm.

*Is the Unwired Server name DNS resolvable?*

Try the Unwired Server name first. If that does not work, the DNS may not be resolvable, Try the IP address instead..

*Is the Unwired Server name reachable through the Internet Control Message Protocol (ICMP)?*

If you cannot ping your Unwired Server name, make sure that you are on the proper network and that your VPN is established and verified as working properly.

*Are Hybrid Web Container ports to Unwired Server going through the firewall?*

Check to see whether the specific ports used by Hybrid Web Container to Unwired Server are going through firewalls. In a Web browser, enter `http://servername:port/tm` (the default port is 5001) to verify the messaging server is listening on the specified port..

*Can an HTTP connection be established?*

If the Hybrid Web Container attempted to connect to the Unwired Server and failed, verify that the HTTP connection is working by going to a website on your device's browser application.

*Check settings in Sybase Control Center*

- Verify that the user name exists on Unwired Server. If it does not, create it in Sybase Control Center.
- Verify that the user has at least one Hybrid App assigned to them. If not, assign a Hybrid App to the user in **Applications > Application Connections**.

- If the device ID is listed under multiple user names, you must remove all but one user name. Search for the Device ID in Sybase Control Center in **Applications > Application Connections**.
- Verify that the user name application ID configuration on the Unwired Server matches the Hybrid Web Container application ID.

## Troubleshoot Hybrid App Development

Provides troubleshooting information for the Hybrid App development and package generation. Hybrid App development can be done using the Hybrid App Designer, Hybrid App API, or a third party independent development environment (IDE).

See the *Validation Rules* section for information about validation rules and error messages.

### Exception Received with Large Amounts of Data

Problem: You receive an exception when attempting to download a large amount of data.

When the amount of data is too large to download to the Hybrid App client device, you may see a message similar to the following in the WorkflowClient trace log:

```
"iAnywhere.MobileOffice.AMP.ResponseRetryException" on device, and following error in WorkflowClient.log:
```

```
20110616T110249.554|4|My:1 -- ExecuteRequest- > Out
20110616T110249.694|1|
iAnywhere.MobileOffice.AMP.ResponseRetryException: Exception of
type
'iAnywhere.MobileOffice.AMP.ResponseRetryException' was thrown.
20110616T110249.694|1| at
Sybase.UnwiredPlatform.WorkflowClient.Utils.CheckMessageSize
(String serializedMessage, ContextData oContextData, Boolean
fromResponder)
20110616T110249.694|1| at
Sybase.UnwiredPlatform.WorkflowClient.Responder.ProcessMessageFromD
evice
(ContextData oContextData, String sData, Boolean isSynchronous)
20110616T110249.694|1| at
Sybase.UnwiredPlatform.WorkflowClient.Responder.ProcessSyncRequest
(ContextData oContextData, String sData)
20110616T110249.694|4|My:1 -- ProcessSyncRequest- > Out
```

Or, when the operation fails, you may see the phrases "COMException" and "Response too large" in the WorkflowClient trace log, or on the client.

Explanation: By default, only messages of 1MB or less are allowed.

Solution: Increase the allowed maximum Hybrid App message size (SupMaximumMessageLength). From Sybase Control Center, expand Hybrid App and select the Hybrid App of interest. Select the Context Variables tab. Modify the property.

See *Configuring Context Variables for Mobile Workflow Packages* in *Sybase Control Center for Sybase Unwired Platform*.

## **Hybrid App Is Not in the Client List**

Problem: The Hybrid App does not appear on the client device.

### *Explanation 1:*

The Hybrid App has not been deployed, or has been deployed, but not assigned to the device.

Solution:

1. Ensure that the user's device has the Hybrid Web Client installed, and that it is configured correctly.
2. Use Sybase Control Center (**Applications > Application Connections**) to ensure that the device is connected.
3. Deploy the Hybrid App, and assign it to the user's device.

### *Explanation 2:*

The Hybrid App does not have a client-initiated starting point.

Solution:

- If you do want to see the Hybrid App in the list, add a client-initiated starting point to the Hybrid App.
- If you realize that the Hybrid App should not be in the list, for example in the case where a Hybrid App only has a server-initiated starting point, no action is required.

### *Explanation 3:*

The device cannot connect to Unwired Server.

Solution:

1. Check the Hybrid Web Client log file for the error `Cannot Connect To Server (503)`.
2. Ensure the settings on the mobile device (**Settings > Connection**) are correct.
3. Ensure that Unwired Server is running, and that the mobile device has connectivity to Unwired Server.
4. If you are using a Windows Mobile emulator, ensure that Active Sync (Windows XP) or Windows Mobile Device Center (Windows Vista, Windows 7) is running. In the Device Emulator Manager, uncradle and recradle.

### *Explanation 4:*

The user attempted to set up a previously connected user on another mobile device, or on a mobile device emulator that has been reset.

Solution:

1. Check the Hybrid Web Client log file for the error `Wrong Device For Code (579)`.
2. Use Sybase Control Center (**Device Users > Devices**) to delete the user and reregister the user.

### Hybrid App Does Not Receive Notifications

Problem: A Hybrid App, which includes a server-initiated starting point, does not receive incoming notifications.

#### *Explanation 1:*

The Hybrid App has not been deployed, or has been deployed but not assigned to the device

Solution:

1. Ensure that the user's device has the Hybrid Web Container installed, and that it is configured correctly.
2. Use Sybase Control Center (**Application > Application Connections**) to ensure that the device is connected and online.
3. Deploy the Hybrid App, and assign it to the user's device.

#### *Explanation 2:*

The matching or e-mail transformation rules are improperly configured, or the e-mail message is improperly formatted.

Solution:

1. Verify that the matching rule and e-mail transformation rules are correct and conform to the e-mail being sent to the device.
2. Send a test notification to the device using the "Send a notification..." option on the Flow Design page of the Hybrid App Designer.

#### *Explanation 3:*

The Notification Mailbox is not properly configured.

Solution:

1. Use Sybase Control Center to verify that the Notification Mailbox is properly configured.
2. Navigate to **Workflows > Notification Mailbox** and ensure that the Enable checkbox is selected.
3. Ensure there is a distribution rule (for example, the To: field must contain the value Custom 1) that matches the properties for the device user (Frank's iOS device's Custom 1 property is set to frank@localhost).

#### *Explanation 4:*

The Hybrid App data change notification (DCN) was not sent correctly.

Solution:



1. Verify that the subsystem responsible for sending the Hybrid App data change notification is functioning properly.
2. Investigate the Unwired Server logs to determine if an error occurred on the server while sending the Hybrid App data change notification.

### *Explanation 5:*

This may happen if another e-mail client is configured and accessing the Hybrid App Notification Mailbox at the same time as the Hybrid App.

Solution: Do not configure other e-mail clients that may intercept the Hybrid App message before it is sent to the Notification Mailbox.

## **Settings Program Does Not Appear on Windows Mobile Device**

Problem: The Settings program does not appear on the Windows Mobile device, after installing the Messaging Client to an actual Windows Mobile device with an ActiveSync connection.

Explanation: This may happen if you installed silently to the device. Install with messages turned on.

Solution: Unselect the Silent Install option, then redeploy the Messaging client.

## **Failed to Create Second Process**

Problem: While trying to view a Hybrid App on the BlackBerry simulator, the operation fails and a message like Failed to Create Second Process appears.

Explanation: This may happen if you attempt to open more than one webkit for the BlackBerry simulator. The BlackBerry simulator has a known limitation of only allowing one webkit instance to run. If you have already opened the browser, then try to open a Hybrid App, you will receive an error like Failed to Create Second Process.

Solution:

1. Close the other webkits.
2. Reopen the Hybrid App.

## **Changes to Custom.js File Do Not Work**

Problem: Changes to the `Custom.js` file do not take effect.

Explanation: This only applies to a Hybrid App written using the Designer.

Solution: After you make changes to the `Custom.js` file, when you regenerate the Hybrid App package, if you select the **Optimize JavaScript in the generated Hybrid App package** option, you must also select the **Update** option in the deployment mode section of the Hybrid App package generation wizard.

## **Hybrid App Package Deployment**

Describes issues with Hybrid App package deployment.

### *Error When Generating a Hybrid App Package*

Problem: When generating the Hybrid App package, you get an error that the Hybrid App package cannot be deployed similar to this:

```
=====
Deployment to Unwired Server
=====
Deploying the workflow
Unable to deploy workflow:
System.Web.Services.Protocols.SoapException: Could not find a
part of the path 'SUP_HOME\Servers\MessagingServer\Data\Mobile
Workflow\117_1'.
at Admin.ReplaceWorkflow(Byte[] baZippedPackage)
```

Explanation: Some software, such as Microsoft Security Essentials, locks the temp folder, or files in the temp folder when they are written by the Hybrid App installation routine, thus preventing the Directory.Move from succeeding.

Solution: Disable Microsoft Security Essentials.

### *Subscription Information Not Removed After Deploying in Replace Mode*

Explanation: Keep in mind that when you deploy a Hybrid App package in Replace mode, Hybrid App assignments are preserved so subscription information is not removed on the server side. If you want to remove the subscription information on the server side, you can remove it manually.

1. Log in to Sybase Control Center.
2. Go to **Applications > Application Connections**.
3. Select the user account, then select **Package Users**.
4. Click **Delete**.

## **Operation Fails with Login Failed Error**

Problem: The operation fails, and `loginFailed:true` appears in the WorkflowClient log.

Explanation: The user name not may not be set properly.

Solution: Make sure that the correct user name and domain (for example, `supAdmin@default`) are specified. Look for `username: supAdmin@default` in the WorkflowClient log.

## **Operation Fails with Package not Found Error**

Problem: The operation fails, and `Package not found` appears in the WorkflowClient log.

Explanation: One or more mobile business objects (MBOs) has not been deployed to Unwired Server.

Solution: Deploy MBOs and retry the operation.

## **Operation Fails with NULL Values Not Supported Error**

Problem: The operation fails, and `NULL <datatype> values are not currently supported in the Sybase Message Client` appears in the WorkflowClient log.

Explanation: The Sybase messaging client cannot differentiate between a NULL value and an empty string. On the device client, there is no way for the user to enter a NULL value for a particular field, so the operation fails.

---

**Note:** This problem occurs in older applications that have not been migrated to Sybase Unwired Platform version 2.0 or later.

---

Solution 1: Modify the MBO definition so that none of its attributes have a NULL value. For example:

```
select emp_id, dept_id, emp_fname, emp_lname, start_date,
termination_date = CASE WHEN termination_date is NULL THEN
'9999-01-01' ELSE termination_date END from employee
```

Solution 2: Add a filter class for the MBO definition to convert the NULL value to a non-null value for attributes.

## **Object Query Listview Does Not Display Data**

Problem: A listview that is populated from an object query does not display any data.

Explanation: The listview key does not match the key being returned from the object query.

Solution:

1. Ensure the appropriate key is selected by clicking the drop-down box for the listview's Key field.
2. Redeploy the Hybrid App.

## **MincheckAffectedRowCountException Error Received**

Possible explanations and solutions for the `MincheckAffectedRowCountException` error.

### *Explanation 1*

If the database data has "\r\n" line endings and if that data is used in an update statement, the data from the Hybrid App control is compared to the database data and they will not match, so an `MincheckAffectedRowCountException` error is thrown. The Document Object Model (DOM) parser converts "\r\n" at the end of line characters to just "\n" when the Hybrid App message is received from the server.

Solution: In the `Custom.js` file, add this code to `customBeforeSubmit` to turn the "\n" back to "\r\n":

```
var idx;
 var values = workflowMessageToSend.getValues();
 var count = values.getCount();
 var keys = values.getKeys();
 for (idx = 0; idx < count; idx++) {
 var data = values.getData(keys[idx]);
 var dataType = data.getType();
 if (dataType === "TEXT") {
 var value = data.getValue();
 if (value.indexOf("\n") > 0) {
 var newValue = value.replace(/\n/gi, "\r\n");
 data.setValue(newValue);
 }
 }
 }
}
```

### *Explanation 2*

The client is specifying outdated values for an update operation, which do not match the values in the back-end data source.

1. In the `WorkflowClient` log, if you see `com.sybase.vader.da.jdbc.MinCheckAffectedRowCountException`, check the preceding line, which should include `JsonContent`. This line contains the old values that are being sent from the client.
2. Check these values against the outdated values in the back-end data source, determine which are different, and make appropriate changes.

## **Hybrid App Push Notification Not Processed**

A Hybrid App push notification is not processed, even though the back-end database is updated with the data.

Explanation: This may happen if another e-mail client is configured and accessing the Hybrid App Notification Mailbox at the same time as the Hybrid App.

Solution: Do not configure other e-mail clients that may intercept the Hybrid App message before it is sent to the Notification Mailbox.

### **PhoneGap API Framework Not Invoked on Device**

On iOS devices, there is no error message, but the PhoneGap framework API is not invoked.

On Android devices, you may receive the error `Error:Can't find the variable:device`.

Explanation: The PhoneGap APIs cannot be accessed successfully until certain initialization has taken place. If you make calls to the PhoneGap API from the `customAfterShowScreen` function, they should occur only after the PhoneGap subsystem is initialized and ready to execute these calls. For more information, see <http://wiki.phonegap.com/w/page/36868306/UI%20Development%20using%20jQueryMobile#HandlingPhoneGapsdevicereadyevent>

### **Hybrid App Stops Working Properly after Upgrade**

Problem: A Hybrid App stops working properly after upgrading to Sybase Unwired Platform.

Explanation: This may be because the Hybrid App is still using PhoneGap libraries.

Solution: Migrate the Hybrid App to use Cordova libraries. See *PhoneGap Library Downgrade* in *Developer Guide: Hybrid Apps*.



# Troubleshoot REST API Development

Provides troubleshooting topics for developing REST services.

## Cannot Access External Services

---

**Problem:** In a network environment with a proxy server, an application cannot access external services through a proxy connection.

**Explanation:** This may indicate the proxy connection or connection pool is not configured properly.

**Solution:**

1. Check the proxy connection pool and verify `EnableHttpProxy` is set to `true`.
2. In Server Configuration (User Options), verify the proxy server settings are included, for example:

```
-Dhttp.proxyPort=8080 -Dhttp.proxyHost=proxy
```

## Application Reports a 401 Unauthorized Error

---

**Problem:** An application reports an "HTTP/1.1 401 Unauthorized" error message.

**Explanation:** This may indicate the configuration is not set up properly.

1. Verify that the URL is correct.
2. Make sure the credentials are valid.
3. Ensure that the security configuration is included in the username (`jd@SC`), or set as the default.

## Security Configuration Users Not Allowed Domain Access

---

**Problem:** An application reports "HTTP/1.1 403 The users of Security Configuration [`<MySC>`] is not allowed to access SUP Domain [`default`]."

**Explanation:** The security configuration may not be assigned to the domain.

**Solution:**

1. In Sybase Control Center, select the **Security** folder under **Domains** > `<DomainName>`.

2. If the required security configuration is not listed, click **Assign** on the right-side **General** pane.
3. Select the security configuration, and click OK.

## **Proxy Connection Not Found**

---

**Problem:** After setting an anonymous proxy connection, an application reports "HTTP/1.1 500 Proxy connection [ ] on Domain [default] not found."

**Explanation:** This may indicate the requested proxy connection is not defined on the requested domain.

**Solution:** Make sure the requested domain is the one defined for the proxy connection. If the requested domain name does not match the host name in the URL, the X-SUP-DOMAIN header can be used to specify the requested domain name.



# Troubleshoot OData Development

Provides troubleshooting topics for developing OData applications in the context of Sybase Unwired Platform.

## Application Reports a 403 Forbidden Error

---

**Problem:** An application reports a "403 Forbidden" error message.

**Explanation:** This may indicate that when a user provided an application endpoint URL while creating the Application ID/ Template, it was automatically whitelisted. But later, when the user tries to change the application endpoint in the application template or application ID, the new URL is not automatically whitelisted.

1. In Sybase Control Center, select the **Domains** folder > **default** > **Connections**.
2. Click **Connections** tab.
3. Click **New**.
  - a) Enter **Connection pool name**.
  - b) Select **Proxy** from **Connection pool type** drop down.
  - c) Enter new application endpoint URL and click **Save**.

## Unable to Delete NoSecLoginModule From the New Security Configuration

---

**Problem:** The NoSecLoginModule still appears even after deletion from Security Configuration.

**Explanation:** This may indicate that when user tries to delete NoSecLoginModule from **Authentication** tab and then click **Apply** in **General** tab, the NoSecLoginModule still appears in **Authentication** tab.

1. Delete the **NoSecLoginModule** from **Authentication** tab.
2. Delete the provider type as **NoSecAuthorizer** from **Authorization** tab.
3. Delete the provider type as **NoSecAttributer** from **Attribution** tab.
4. In **General** tab, click **Apply**.

## Native Push Notification Not Working After ODP Upgrade

Problem: Native push notification does not work after Unwired Server is upgraded to a later version.

Explanation: Native push notification is not working properly for the same application ID when the Unwired Server is upgraded to the newer version.

- Check the format of push endpoint URL. For example:  
`http://<server address>:8000/notifications/v1`
- The administrator should manually change the preexisting push endpoint URL after upgrade.
- For newly created application IDs/ application template (after upgrading) it automatically considers the new push endpoint.

## Application Reports a 401 Unauthorized Error

Problem: An application reports an "HTTP/1.1 401 Unauthorized" error message during request response in anonymous registration.

Explanation: This may indicate the connection pool is not set up properly.

Solution:

- Check that the connection pool is created with the same name as the Application ID.
- Check that the appropriate URL and user credentials are provided in the connection pool. If not, then:
  - a) Use the same URL as mentioned in Application ID.
  - b) In the connection pool, add property to **User**, **Password** and provide the gateway credentials.
  - c) Ensure the **Allow Anonymous Access** property is set to **True**.

# Troubleshoot Agency Development

Provides troubleshooting topics for developing Agency applications in the context of Sybase Unwired Platform.



# Troubleshoot Device Clients

Provides troubleshooting information for device client applications that are typically deployed in the field.

## **Device Provisioning**

---

Provides troubleshooting information for provisioning device application clients using Afaria.

### **Device Cannot Communicate with Afaria Server**

---

**Problem:** When Afaria is part of the same mobile enterprise as Unwired Platform, Afaria Server appears to not respond to incoming requests from Unwired Platform components.

**Solution:**

- Ensure that Afaria Server is running.
- Check whether the device browser can open Web pages correctly to determine whether device network access is functioning.

### **Problem Loading .cod Files on the BlackBerry Device**

---

When trying to load the .cod files for a Hybrid Web Container or BlackBerry application on the BlackBerry device you receive an error saying "The following modules could not be loaded because their digital signatures were invalid."

**Explanation:** The IT policy on the BlackBerry device might be preventing you from loading the necessary .cod files.

**Solution:** Ask your BES Administrator to modify the IT Policy for your device, to enable the installation of third-party applications.

## **Device Clients**

---

Provides troubleshooting information for device application clients. Some topics may apply to device application clients in general, while others may be specific to platform types.

### **No Workflow Message is Sent**

---

**Problem:** After a new hybrid web container has been installed on a device, attempts to connect the device to Unwired Server or to send a data change notification (DCN) result in an error: No

workflow message is sent because no device is found to send the notification.

Explanation: When a new application and application connection are created and registered from Sybase Control Center, by default there are no users associated with a device. Once a user invokes the workflow from the device, the device user binds to the application connection, and the device user appears in the Application User tab in Sybase Control Center.

In this case, the application and application user connection failed to bind when the user invoked the workflow from the device, resulting in a message similar to:

```
{"id":"dcntest_test_001","success":false,"statusMessage":
"no workflow message is sent because no device is found to send the
notification"
,"result":
[{"id":"dcntest_test_data_1","success":true,"statusMessage":""}]}
```

Solution: To initiate the binding manually, instruct the user to invoke a workflow from the device, or run an MBS application on the device. This action should invoke a certain MBO query, which requires a Sybase Unwired Platform user.

1. Invoke any other Unwired Platform application (for example, a client-initiated or server-initiated workflow application, or a native application), which maps an Unwired Platform application connection to an MBS user.
2. Send the DCN with to ":" < *this SUP user* >.

### **Synchronization Fails in a Remote Server Environment**

Problem: Synchronization fails due to an error on the server.

Explanation: This may happen if you configured `sampledb` in a remote server environment, and accepted the default server name "localhost." The default server name causes synchronization to fail.

Solution: Ensure that the `sampledb` server name reflects the exact IP number or server name of the remote server:

1. Launch Sybase Control Center, and, in the Administration perspective, open the remote cluster.
2. In the left navigation pane, expand the appropriate domain, and click **Connections**.
3. In the **Connections** tab, select **sampledb**, and click **Properties**.
4. In the Properties dialog, edit the `sampledb` server name value to reflect the appropriate remote server name or IP address.

### **Synchronization Fails With OutOfMemory Error**

Problem: The device application seems to hang for 15 or 20 minutes, then synchronization fails with a `java.lang.OutOfMemoryError` or `OutOfMemoryError` message.

Explanation 1: The load and synchronization parameters may not be set correctly to eliminate extraneous data. The volume of synchronized data, both from the enterprise information

system (EIS) to the cache database (CDB), and from the CDB to the device client is causing the out-of-memory error.

Solution: Redesign the mobile business object (MBO) to take better advantage of both the load and synchronization parameters. Search *Sybase Unwired WorkSpace - Mobile Business Object Development* for the topic *Combining Load and Synchronization Parameters* to learn more.

Explanation 2: The device database page size may be too small.

Solution 1: Increase the page size for the device, using the Sybase Unwired WorkSpace code generation wizard. Search *Sybase Unwired WorkSpace - Mobile Business Object Development* for the topic *Generating Object API Code*, or search the appropriate *Developer Guide* for information about generating code for a specific device platform.

## **Synchronization Fails With Exception**

Problem: Some applications fail to synchronize, and report either an uncaught exception for `waitForKeyGenerator` when calling the **subscribe** method, or an `OutOfMemoryException`.

Explanation: This may happen if Unwired Server does not send the key generator row in response to a login request from the client within 20 seconds of the client's subscribe request. This is typically because the mobile business object is handling a large amount of data. It can also happen if there is a connection problem with the enterprise information system (EIS) that slows response time.

Solution: Try one or more of these:

- Increase the Unwired Server `Max Heap Size`. Search for the *Configuring System Performance Properties* topic found in both *System Administration* and the *Sybase Control Center* online help.
- Redesign the mobile business object (MBO) to handle less data in a transaction.
- Develop custom logging to better trap error messages for more details.

## **Timeout Errors During Client Synchronization**

Problem: Various timeout errors are reported during client synchronization, especially in a cluster configuration.

Explanation: This may indicate that performance tuning is required to make Unwired Server processing more efficient.

Solution: Take one or more of the following actions to improve performance.

### *Action 1: Review the Configuration*

Review the cluster configuration for settings that may impact performance.

## Troubleshoot Device Clients

1. Monitor the CPU utilization during client synchronization to determine the percent used. Utilization should be less than 80% for the cluster; ideally no more than 70 – 80%. If it is higher, check the Unwired Server logs for related messages.
2. High CPU utilization may indicate the cluster is bound by physical I/O (the disk queue is over utilized). The administrator may check the performance monitor to confirm this state. When the disk queue is backed up, requests may start to time out and the overall system throughput will degrade. You may need to reduce the load on the cluster by decreasing the number of MobiLink threads on each Unwired Server. See *System Administration*, the *Performance Tuning and Testing* section.

### *Action 2: Reconfigure Client Connections*

Configure the client connections to use a longer timeout on the connection. The default is the minimum value of 240 seconds, and the maximum value is 600 seconds. This value is set by adding the timeout property and desired value to the NetworkStreamParams on the connection.

For CS/UltraLite, assuming the generated DB class name is "MyPackageDB", the code to set timeout could be:

```
MyPackageDB.GetSynchronizationProfile().NetworkStreamParams="trusted_certificates=;url_suffix=;timeout=600"
```

For Java/UltraLiteJ, you cannot set timeout, because UltraLiteJ does not provide an API to set a timeout value for client connection. Sybase Unwired Platform generated client code for UltraLiteJ (such as for Java desktop and RIM replication based synchronization client) only supports the "trusted\_certificates" and "url\_suffix" properties in NetworkStreamParams. If you set other properties in NetworkStreamParam, they will be ignored.

For additional information on setting the NetworkStreamParams properties, search *System Administration* for references to "stream parameter." The topic *Verifying Device Client HTTPS Setup* provides this sample code for setting stream parameter:

```
"url_suffix=/ias_relay_server/client/rs_client.dll/[SMP_FARM_ID];tls_type=RSA;trusted_certificates=rsa_root.crt;identity=id_client.pem;identity_password=pwd;"
```

### *Action 3: Modify Client Application Code*

Add retry logic to your client application code to retry syncs that fail with timeout.

## **Device Operation Fails With SQLE\_NOT\_IMPLEMENTED Error**

**Problem:** A Windows Mobile or BlackBerry device throws an Exception with the message SQLE\_NOT\_IMPLEMENTED.

**Solution:** If the device client database throws the exception SQLE\_NOT\_IMPLEMENTED in a JOIN query, check your join fields to make sure they are the same type.



## **No Data Displays on Device Client**

Problem: No data displays on an iOS or Windows Mobile device, and no error message displays.

Explanation: This may happen because of a subscription failure. Because of the failure, no data is downloaded to the device. Possible causes:

- iOS – the DOE Connector Listener URL is not correct.
- Windows Mobile - the client is not connected to the messaging service (the user displays as offline in the Sybase Control Center administrative console). The DOE Connector ESDMA package did not deploy properly (especially if the Listener URL is not correct).

Solution:

1. Verify that the DOE Connector Listener URL is correct.
2. Redeploy the DOE Connector ESDMA package.

See:

- *Sybase SAP DOE Connector Installation Guide*
- *System Administration for Sybase Unwired Platform > System Reference > EIS Data Source Connection Properties Reference > SAP DOE-C Properties*
- *Sybase Control Center > Configure > Configuring Unwired Platform > Packages > Configuring Messaging and DOE-C Package Device Settings*

## **Synchronizing the Device Application Returns Only One Row of Data**

Problem: If the enterprise back-end database table has no primary key and contains multiple rows of identical information in every column, a device application user can synchronize only one row.

Explanation: If a device application user tries to create multiple rows that are identical in every column, it succeeds to update the back-end database, but only displays one column on the client and the cache database (CDB).

Solution: verify the back-end database has a primary key.

## **Packet Dropped State**

Problem: User stops receiving packets for a package subscription and "Packet Dropped" displays as the subscription status for the user and device ID.

Explanation: A problem has occurred in processing the packet. No further packets sent to subscriptions for this package will be processed until the problem is fixed.

Solution:

1. In the Unwired Server log, locate the lines recording the packet drop.

- a. In a text editor, open `<SUP_installdir>\Servers\UnwiredServer\logs\<server_name>-server.log`.
- b. Search for "packet-dropped" or "packet dropped" – you should find two consecutive lines that show this text far to the right:

```
... Detected a packet-dropped candidate message ...
... DOE-C is negotiating a DOE packed dropped message:
```

2. Determine the reason that the packet was dropped.

- a. If you see this line below following close after the two lines above, the packet was dropped because the packet drop size was exceeded:

```
MESSAGE: Request Entity Too Large
```

Look further to the right in the first line above and note the "size" value and the "maximum" value:

```
... candidate message - size ####, maximum ####
```

- b. If you see this line below following close after the two lines above, the packet was dropped because the packet read timed out – the packet could not be processed in the time allotted:

```
Read timed out : Read timed out
```

- c. If you see neither of the lines above following close after the first two, the packet was probably dropped because it could not be parsed correctly.

3. Use the table below to finish resolving the issue.

Reason for drop	Next action	Resolution options
Packet drop size exceeded	Check with your SAP administrator to see if the packet "size" value can be made smaller than the "maximum" value (the packet drop size).	<p>If your SAP administrator can reduce the packet size below the packet drop size:</p> <ol style="list-style-type: none"> <li>1. Have the SAP administrator can reduce the packet size.</li> <li>2. Unsubscribe and resubscribe all affected user devices.</li> </ol> <p>If your SAP administrator cannot reduce the packet size below the packet drop size:</p> <ol style="list-style-type: none"> <li>1. Verify that user devices can handle a packet as large as the one that was dropped.</li> <li>2. In Sybase Control Center, increase the packet drop size to accommodate the size of the dropped packet: select the package, go to the <b>Subscriptions</b> tab, click the <b>Properties</b> button.</li> </ol>

Reason for drop	Next action	Resolution options
Packet read timed out	Contact your network administrator.	<p>Have your network administrator verify that the timeout was not caused by an unrelated network problem. For example, if the DOE server was unable to resolve the hostname of the Sybase Unwired Server, or a network router failed, a read timeout would result.</p> <p>If you have ruled out network problems as the cause of the timeout, have your SAP administrator increase the packet read timeout value for the package on the SAP end.</p> <p>Then, in Sybase Control Center, increase the packet read timeout value for the package: under <b>Domains</b>, select <b>default, Connections</b>, select the package, then click <b>Properties</b> – DOE SOAP Timeout is the packet read timeout setting.</p>
Packet could not be parsed correctly	Contact your SAP administrator.	Have your SAP administrator fix the parsing issue for the package on the SAP end.

For more information on:

- **Understanding packet drop size** – see *System Administration for Sybase Unwired Platform > System Reference > EIS Data Source Connection Properties Reference > SAP DOE-C Properties*
- **Setting packet drop size and read timeout in Sybase Control Center** – see *Sybase Control Center Online Help > Configure > Packages > Viewing and Changing Messaging and DOE-C Package Connection Properties*
- **Setting packet drop size and read timeout with the Command Line Utility** – see *Installation Guide for Sybase SAP DOE Connector > SAP DOE Connector Command Line Utility > Package Management Commands > setEndpointProperties Command.*

### Pattern is Too Long Exception

**Problem:** An exception—The pattern is too long—is reported while trying to update an existing value on a back-end database, via mobile device.

**Explanation:** This case may happen when a mobile business object's (MBO's) operation definition is generated by dragging and dropping a database table to the object diagram in Unwired Workspace.

The error is caused by an incorrect LIKE clause. The LIKE clause should not be used by a text data type whose length is greater than 254 characters, or a binary data type. The **LIKE** clause is

## Troubleshoot Device Clients

the Unwired WorkSpace's best effort to resolve operations executed against tables that do not contain a primary key.

Solution: Redesign the back-end database to contain a primary key on the offending table.

### **Feature Not Implemented Exception**

Problem: An INSERT operation on the client device results in an exception similar to:  
Feature 'n/a' not implemented.

Explanation: This may indicate that page size is not large enough to accommodate all MBO attributes included in the mobile application project.

The page size should be larger than the sum of all attribute lengths for any MBO which is included with all the MBOs selected, and must be valid for the database. If the page size is changed, but does not meet these guidelines, object queries that use string or binary attributes with a **WHERE** clause may fail.

Solution: Regenerate the code, this time specifying a larger page size. For information search *Sybase Unwired WorkSpace - Mobile Business Object Development* for the topic *Generating Object API Code*, or search the appropriate *Developer Guide* for information about generating code for a specific device platform.

### **Synchronization Key Rows Must be Unique**

Problem: For device clients connecting to DOE, duplicate usage of synchronization keys on inbound create, read, update, or delete (CRUD) messages from device clients can cause unexpected behavior on the DOE side.

Solution: Ensure that synchronization keys are globally unique identifiers (GUID) and that every single row is unique.

## **BlackBerry**

Provides troubleshooting information specifically for BlackBerry device application clients.

### **Uncaught exception:java.lang.Error**

Problem: You receive an uncaught `exception:java.lang.Error` when starting a device application on a BlackBerry device.

Solution: Verify that the mobile application package containing the mobile business objects referenced by the device application has been deployed to Unwired Server.

### **BlackBerry Device Login Error**

Problem: The BlackBerry device reports login errors for various situations.

Explanation: The login error may be reported for various situations, in addition to login problems.

Solution: Review the Unwired Server log to get the exact error.

### **BlackBerry Device Synch Fails**

Problem: BlackBerry device synchronization fails, and the Unwired Server log reports a Cannot synchronize message.

```
Thread-141 [com.sybase.ml.sup.Logger] [-10013] Version
'default_flightmodelsample.1_0'
not found in the ml_script_version table. Cannot synchronize
```

Explanation: A package type mismatch can cause synchronization errors; for example, if the device synchronizes an RBS package with a deployed MBS package, the synchronization will fail.

Solution: Redeploy so the package types match.

### **BlackBerry Device Client Cannot Sync**

Problem: The BlackBerry device client cannot synchronize and reports a SynchronozationException message that indicates a `SOLE_COMMUNICATIONS_ERROR`.

Explanation: For applications that use replication-based synchronization (RBS), this may indicate the Unwired Server is down.

Solution: Report that Unwired Server is down, or wait for it to come back online.

### **BlackBerry Device Cannot Find Valid Certification Path**

Problem: The BlackBerry device client cannot set up a secure connection and reports a `ASynchronizationException` message that indicates a certification problem.

```
Communication error sun.security.validator.ValidatorException: PKIX
path building
failed: sun.security.provider.certpath.SunCertPathBuilderException:
unable to find valid
certification path to requested targetDetails:
```

Explanation: One of the most common causes for not being able to set up a secure connection is an invalid certificate.

Solution: Make sure the certificate is valid, and the secure connection is set up correctly.

### **BlackBerry Simulator Fails to Load URL**

Problem: In your application, when you attempt to open a URL on an older BlackBerry 6.0 simulator, the URL fails to load, and the sim the error message 'Failed to create a second process.'

Explanation: The `showUrlInBrowser` method does not work on older BlackBerry 6.0 simulators.

Workaround: Use a later version of the simulator, such as 6.0.0.534.

### **Value INTEGER Out of Range For Destination Error**

**Problem:** When you enter a value on a BlackBerry device in an attribute of the BYTE type, you receive this error when submitting the data.

**Explanation:** This error can happen if the mobile business object is associated with a SQL Anywhere database. With SQL Anywhere, the `tinyint` column is automatically mapped to be an unsigned TINYINT type attribute, which can hold values from 0 to 255. However, if you manually map the `tinyint` column to a BYTE attribute, on BlackBerry devices, only values from 0 to 127 can be used.

### **Zero Value Returned Instead of Personalization Key Value**

**Problem:** A zero value was returned on a BlackBerry device instead of the expected personalization key value.

**Explanation:** This is expected behavior if a default value is not set for a non-nullable personalization key. When a default value is not set for a non-nullable numeric personalization key, the object API can only return a zero for `getXXX` methods.

**Solution:** If a default value is not set, the device user must set the personalization key value manually. Alternatively, redesign the application to include the default value.

### **Date Time Picker Pops up on BlackBerry Simulator**

On BlackBerry 9800 Simulators, the native datetime/date/time picker always pops up, even when clicking the datetime/date/time type Read-only edit box in the detail screen.

**Explanation:** When the disabled datetime/date/time HTML field is clicked, the native datetime/date/time picker pops up. This only happens on the BlackBerry simulator, not on BlackBerry devices.

**Solution:** None. Do final testing on a BlackBerry device to confirm.

### **OutOfMemory Error on BlackBerry Simulator**

**Problem:** Received an OutOfMemory error on the BlackBerry simulator after inserting a graphic image, saving, and submitPending in the client.

**Explanation:** Assuming the mobile business object (MBO) is set properly with a single binary attribute, this can mean the size of the binary attribute is not sufficient. You may need to expand its size.

**Solution:** Reset the simulator default heap size to a larger value. You may need to experiment with the BlackBerry simulator you are using. For example, you might develop guidelines such as these after experimenting with a BlackBerry 9550 simulator:

- If you set the simulator heap size to 100MB – the suggested size of the binary attribute is 2.5MB. Beyond that size you will probably see an OutOfMemory error in the client.
- If you set the simulator heap size to 256MB – the suggested size of the binary attribute is 6MB. Beyond that size you will probably see an OutOfMemory error in the client.

If an MBO has more than two binary attributes, Sybase recommends that their combined size not exceed the limit. The OutOfMemory error might not occur if you exceed the limit, but it is a suggested guideline to prevent exceeding the limit.

### **Server Response Timeout Error on BlackBerry Simulator**

**Problem:** After restarting the BlackBerry client simulator, received a Server Response Timeout error in the event status log (Sybase Setting > Show Log).

**Explanation:** This typically indicates a problem within the BlackBerry MDS simulator environment. For example, the timeout setting in the BES/MDS configuration may be set too low for Unwired Server.

**Solution:** Restart the MDS and device simulator frequently to start fresh.

### **Server Response Timeout Error on BlackBerry Device**

**Problem:** The BlackBerry device client shows a Server Response Timeout error in the event status log (**Settings > Show Log**).

**Explanation:** In general, a "Server Response Timeout" error means that the server did not respond to an HTTP request in a timely manner. This typically indicates that the request was too large or the network too slow to complete a request before either the BES or the network layer on the client timed it out. This could be caused by the BES/MDS configuration being set to a timeout that is too short for the kind of requests being sent (for example, very large uploads or downloads). Additionally, a slowly responding Unwired Server (perhaps because it is overloaded) could cause the same error.

**Solution:** Check Unwired Server, the network, and BES/MDS configuration settings, and tune for better performance. See *System Administration*, the *Performance Tuning and Testing* section.

### **Workflow Messages Not Delivered to Inbox**

**Problem:** You only receive workflow messages in the inbox after restarting the client. The device shows 'Online' in SCC, and the Show Log message "Awaiting Server Notification" appears. Only after you restart the client is the message delivered to the inbox.

**Explanation:** The problem may be caused by having the Unwired Server installed on one machine or VM, and the BlackBerry and MDS simulator running on another machine or VM.

**Solution:** Change the BlackBerry Push Configuration in Sybase Control Center to point to the machine or VM running the BlackBerry MDS simulator, rather than the default configuration of localhost:8080.

### **Workflow Application Takes a Long Time to Open on BlackBerry 7.0 and Later**

**Problem:** The Workflow application takes a long time to open.

**Solution:** When you generate the Mobile Workflow Package, selecting the **Optimize JavaScript in the generated workflow package** option in the wizard can make the Workflow open more quickly.

### iOS

Provides troubleshooting information specifically for iOS device application clients.

#### **iOS Device Synchronization Fails**

**Problem:** iOS device synchronization fails, and the Unwired Server log reports a `subscribeFailure` message.

**Explanation:** A package type mismatch can cause synchronization errors. The `subscribeFailure` message imports a log record to the client with the reason for the failure (such as package not found, or invalid credentials). You can use the database class method **`getLogRecords`** to get the record that was imported.

**Solution:** Use message information to solve the problem. For example, you may need to provide valid credentials, correct settings (for example, only MBS is supported for iOS), or reinstall the package on Unwired Server.

#### **iOS Device Cannot Sync**

**Problem:** The device client cannot synchronize with Unwired Server, with errors.

**Explanation 1:** The Unwired Server may be down. The error would report an `NSEException` that includes `name: loginToSync`, and `reason: No connection to server because the Unwired Server is down or unreachable`. The `loginToSync` method can detect if the server is unreachable (the messaging layer is not in "Connected" state) within a timeout.

---

**Note:** Sybase does not recommend using `loginToSync`, because it blocks while waiting for network response. Instead, use `beginOnlineLogin`. This call sends the login message to Unwired Server and returns immediately. The application is then responsible for detecting login success or failure using the callback handler.

---

**Solution:** Report that Unwired server is down, or wait for it to come back online. Alternatively, redesign using `beginOnlineLogin`.

**Explanation 2:** The connection between the device and Unwired Server may be down. You can test to make sure the device is connected as required using the `SUPApplication` API:

```
// make sure connection established
while ([SUPApplication connectionStatus] !=
SUPConnectionStatus_CONNECTED) {
 NSLog(@"waiting to connect...");
 sleep(2);
}

NSLog(@"connection established");

while ([SUPApplication registrationStatus] !=
SUPRegistrationStatus_REGISTERED) {
 NSLog(@"waiting to register...");
```



```

 sleep(2);
 }

 NSLog(@"registration completed");

 [SampleApp_SampleAppDB beginOnlineLogin:@"supUser"
password:@"s3pUser"];

 while([SampleApp_SampleAppDB getOnlineLoginStatus].status ==
SUPLoginPending)
 {
 [NSThread sleepForTimeInterval:0.2];
 }
 // After this, the status will be either SUPLoginSuccess or
SUPLoginFailure
 if([SampleApp_SampleAppDB getOnlineLoginStatus].status ==
SUPLoginSuccess)
 [SampleApp_SampleAppDB subscribe];

```

Solution: This reports the connection status. With a connection established, run this code on the device. After you run the code on the device, you can check from Sybase Control Center to see if the device is online.

### **Repeating a Query Produces No Results**

Problem: On an iOS device, you cancel a query for a mobile workflow message, then repeat the same query. The second query does not produce results, and you see a message similar to Error value is null or empty.

Solution: Redesign the client application using the Mobile Workflow Forms Editor to prevent users from navigating back in the client application.

### **Localized Container Application Displays English**

Problem: Your localized container application displays in English, rather than the target language.

Explanation: You have set the local and language set to your target language, but instead see English displayed in the container application.

Solution: Check to make sure the correct region is set by selecting **International > Region Format** on the iOS device.

## **Windows Mobile and Windows**

Provides troubleshooting information specifically for Windows Mobile and Windows device application clients.

### **Windows Mobile Device Synchronization Fails**

Problem: Windows Mobile or Windows device synchronization fails.

Explanation: If the device synchronizes a replication-based synchronization (RBS) package with a deployed MBS package, the synchronization fails. The Unwired Server log will contain an error message indicating that the script version is not found.

```
Thread-141 [com.sybase.ml.sup.Logger] [-10013] Version
'default_flightmodelsample.1_0'
not found in the ml_script_version table. Cannot synchronize
```

If the device uses message-based synchronization (MBS) mode to subscribe to a deployed RBS package, the device application receives a 404 error indicating "Package not enabled for message-based synchronization."

Solution: Redeploy so the package types match.

### **Windows Mobile Device Cannot Sync**

Problem: The Windows Mobile or Windows device cannot synchronize. Additional error messages and behavior differ for replication-based synchronization (RBS) and message-based synchronization (MBS).

Explanation: This may indicate the Unwired Server is down or unreachable.

- For RBS clients, if the device attempts to synchronize with the Unwired Server but the Unwired Server is down, the device receives a `SynchronizationException` indicating a `SOLE_COMMUNICATIONS_ERROR`.
- For MBS clients, the Sybase Messaging Client will keep retrying the request. The request will be handled when the Unwired Server restarts.

Solution: Report that Unwired Server is down, or wait for it to come back online.

### **Windows Mobile Device Cannot Establish Secure Connection**

Problem: The Windows Mobile or Windows device client cannot set up a secure connection and reports a `SynchronizationException` message that indicates a communication error.

Explanation: One of the most common causes for not being able to set up a secure connection is an invalid certificate.

Solution: Make sure the certificate is valid, and the secure connection is set up correctly.

### **Cannot Test Database File on SD Card from Simulator**

Problem: Cannot test a database file that is stored on an SD card from the Windows Mobile simulator.

Explanation: This is a known limitation of the Windows Mobile simulator. The simulator cannot test files on the SD card.

Solution: If you want to test the database file on the SD card, you must use a real device.

**Failed to Create Component TextBox**

Problem: Failed to create a component TextBox using drag and drop for a Windows Mobile client application. Error messages provide additional information.

```
Failed to create component 'TextBox'. The error message follows: 'System.IO.FileLoadException: Could not load file or assembly 'Microsoft.WindowCE.Forms, Version=3.5.0.0,Culture=neutral, PublicKeyToken =...'
```

Explanation: This indicates you must use a Windows command to complete the action.

Use the Windows command `gacutil /i Microsoft.WindowCE.Forms.dll` to add the assembly to the cache.

**Import Fails with OutOfMemory Error**

Problem: While importing data for large columns on Windows Mobile, import for one large column succeeds, but import for an additional large columns fails with errors.

```
YYYY-MM-DD HH:MM:SS OnMessageException: malloc() failed
out of memory at System.Data.SQLite.SQLite3.Reset(SQLiteStatement
stmt)
 at System.Data.SQLite.SQLite3.Step(SQLiteStatement stmt)
 at System.Data.SQLite.SQLiteDataReader.NextResult()
 at System.Data.SQLite.SQLiteDataReader..ctor(SQLiteCommand cmd,
CommandBehavior behave)
 at System.Data.SQLite.SQLiteCommand.ExecuteReader(CommandBehavior
behavior)
 at System.Data.SQLite.SQLiteCommand.ExecuteNonQuery()
 at com.sybase.afx.db.SQLiteBigStringImpl.OpenForWrite(Int64
length, Boolean createPending)
 at end2end.rdb.Author.DealBigAttribute(JsonStreamParser _parser)
 at end2end.rdb.Author.StreamImportHandler(JsonStreamParser
_parser)
 at end2end.rdb.intrnl.Author_ML.OnMessage(JsonMessage msg)
 at end2end.rdb.End2end_rdbDB.DispatchReceivedMessage(JsonMessage
message)
 at end2end.rdb.intrnl.End2end_rdbDB_ML.OnMessage(JsonMessage msg)
 at
com.sybase.sup.client.mbs.QueueConnectionImpl.OnMessage(JsonMessage
message)
 at
com.sybase.sup.client.mbs.SupHandler.HandleAsyncMethodCallFromServe
r(String headers, MoBinary data)
 at com.sybase.sup.client.mbs.SupMocaObject.Run(String
strMethodName, MoParams oParams)
 at Sybase.MessagingClientLib.MocaObject.OnMocaRun(String
sMethodName, IntPtr oParamHandle)
```

Explanation: This issue is specific to Windows Mobile DOE-C based applications (message based synchronization) that use the SQLite database to store data on the device. SQLite has a

## Troubleshoot Device Clients

row size limitation of 4 - 5MB. The Mobile Business Object (MBO) cannot execute an operation because the row size limit has been reached.

Solution: None. You cannot work around this limitation from a code point of view. You may be able to solve the problem by reducing the amount of data retrieved by the MBO.

## **Android**

Provides troubleshooting information specifically for Android device application clients.

### **Client Failed to Initialize Error**

Problem: After registering, the Android device client shows a Client Failed to Initialize error in the messaging log.

Explanation: The message is not accurate and can be ignored.

# Troubleshoot Command Line Utilities

Provides troubleshooting information for Sybase Unwired Platform command line utilities.

## Utilities Cannot Locate djc-setenv.bat

---

**Problem:** Environment variables are frequently defined in the setup `djc-setenv.bat`. However, if you do not run these utilities from `<UnwiredPlatform_InstallDir>\servers\unwiredserver\bin`, Unwired Platform command line utilities cannot find this setup script. This causes the command line utility to not run as expected.

**Solution 1:** Remember to manually change to the `bin` folder before running the utility.

**Solution 2:** Use the following command line syntax to add a `CD` command in the batch file to change the directory before calling the utility:

```
"cd <UnwiredPlatform_InstallDir>\servers\unwiredserver\bin &
<UnwiredPlatform_InstallDir>\servers\unwiredserver\bin
\utilityName.bat
```



# Validation Rules



Validation rules ensure that mobile business objects (MBOs) and the device applications and lightweight mobile workflow applications they interact with perform as expected in an enterprise environment. Error messages report problems and are useful for diagnosing the problem.

## Mobile Business Object Validation Rules and Error Messages

---

Unwired WorkSpace (Eclipse) mobile business object (MBO) validation rules and error messages are divided into sections based on where in the MBO development cycle the validation rule applies.

Some errors and warnings are generated when you violate a validation rule, while others may occur during code generation, deployment, or when used within the device application, in which the error might not be obvious. Error information includes:

- Severity and message – there are two types of messages generated if a validation error occurs:
  -  Warning messages – A nonfatal error that requires attention, or a failure may occur at a later time.
  -  Error messages – the operation being performed fails. You must resolve the error.
  - Message - the actual error message, if any, that is generated if the validation rule is violated.
- Action – any additional details regarding the validation rule and how to resolve the error.

---

**Note:** You can identify some validation errors in Unwired WorkSpace by the warning or error icon that appears within the context of the error. For example:

- In the Properties view, if you add an unmapped parameter to an attribute, a warning icon and message appear. The same warning icon also appears in the header of the MBO in the Mobile Application Diagram.
- Mobile application project folders in WorkSpace Navigator display warning or error icons for invalid objects within the folder (MBOs, roles, personalization keys, and so on). You can isolate the error by expanding the folder until you reach the object with the error.

Other types of errors, for example deployment errors, generate messages.

## Validation Rules

In general, be aware of warning and error icons and messages, and resolve them early in the development cycle to ensure proper deployment, code generation, and device application function.

---

### **Mobile Business Object, Attribute, and Structure Object Error Messages and Validation Rules**

These validation rules and error messages apply to mobile business objects (MBOs), attributes, and structure objects (complex object types), and are enforced during MBO development, and, in some cases, when you deploy the MBO to Unwired Server. Messages are also generated when validation rules are violated.

**Table 7. MBO, attribute, and structure validation rules and error messages**

<b>Severity and message</b>	<b>Cause</b>	<b>Action</b>
Error: Invalid mobile business object name: '<MBO Name>'. A valid identifier name must begin with an alphabetic character or an underscore, followed by alphanumeric characters or underscores.	The MBO name is invalid.	Enter a valid name.
Error: Invalid mobile business object name: '<MBO Name>', name cannot be a Java or C# language keyword.	The MBO name is a Java or C# keyword, such as binary, boolean, byte, char, date, datetime, time, decimal, double, float, object, int, integer, long, short, string, switch, and so on.	Do not use Java or C# keywords as MBO names.
Error: Invalid mobile business object name: '<MBO Name>', the length must be less than 64.	The length of the MBO name is longer than 64 characters.	Enter a name that is less than 64 characters.
Error: Invalid mobile business object name: '<MBO Name>', name cannot be a reserved attribute name.	The MBO name is a reserved attribute name: pending, replayPending, replayCounter, disableSubmit, pendingChange, lastModified, and deleted.	Enter a valid name.



Severity and message	Cause	Action
Error: Invalid mobile business object name: '<MBO Name>', name cannot be a reserved word.	The MBO name is a reserved MBO name: KeyGenerator, OperationReplay, ServerPersonalization, ClientPersonalization, SessionPersonalization, LogRecordImpl, and KeyGeneratorPK.	Enter a valid name.
Error: Invalid mobile business object name: '<MBO Name>', name cannot be a reserved internal type name.	The MBO name is an internal reserved type: "object", "binary", "boolean", "byte", "char", "date", "datetime", "decimal", "double", "float", "int", "integer", "long", "short", "string", "time"	Enter a valid name. For example, naming the MBO Date, DateTime, or Time results in an error when deploying the MBO.
Error: Duplicate mobile business object name: '<MBO Name>'. The name is case-insensitive.	An MBO with the same name already exists.	Enter a valid and unique name.
Error: Duplicate name: '<MBO Name>' in the object query.	The name of the MBO is the same as the name of an existing object query.	Enter a valid and unique name.
Error: The name: '<MBO Name>' duplicates with an operation name.	The name of the MBO is the same as the name of an existing operation. Validation occurs after creating the MBO after a table name change to an operation or attribute, then dragging-and-dropping the table to create the MBO.	Enter a valid and unique name.
Error: The name: '<MBO Name>' duplicates with an attribute name.	The name of the MBO is the same as the name of an existing attribute.	Enter a valid and unique name.
Error: Invalid structure name: '<Structure Name>'. A valid identifier name must begin with an alphabetic character or an underscore, followed by alphanumeric characters or underscores.	The structure type name is invalid.	Enter a valid name.

## Validation Rules

Severity and message	Cause	Action
Error: Invalid structure name: '<Structure Name>'. A valid structure type name cannot be the same as the primitive type names.	The structure type name is a primitive datatype supported by Unwired WorkSpace. For example, int, boolean, and so on.	Enter a valid name.
Warning: Neither attribute nor operation is defined for the mobile business object: '<MBO Name>'.	An MBO exists without attributes or operations.	Define attributes, operations, or both for the MBO.
Warning: Attributes are not defined for the structure: '<Structure Name>'.  	A structure exists without attributes.	Define attributes for the structure.
Error: Attributes are not defined for the mobile business object: '<MBO Name>'.  	This error occurs in the MBO Creation wizard, in the "Attributes mapping" page if the wrong datasource operation definition is specified on the "Definition" page. In these cases, the data source operation cannot run successfully or there is no output record for the operation. No attributes are generated for the MBO.	In the MBO Creation wizard, do one of: <ul style="list-style-type: none"> <li>• Click <b>Back</b> to return to the "Parameters" page and verify that the correct default values are set. Correct any values as necessary then click <b>Next</b> to retry.</li> <li>• Click <b>Back</b> until you reach the "Definition" page and verify that the correct definition for the datasource operation is specified. Correct the problem and click <b>Next</b>, and specify default values for the parameters (arguments) on the "Parameters" page. Click <b>Next</b> to retry.</li> </ul>
Warning: Attributes are not bound to the back-end data source: '<MBO Name>'.  	You can create an MBO without specifying a datasource, by selecting <b>Bind data source later</b> from the "Data source" page of the MBO Creation wizard. This warning message reminds you to bind the MBO to a data source at some point during the development process.	To bind an MBO to a data source after creating the MBO: <ol style="list-style-type: none"> <li>1. Right-click the MBO in the Mobile Application Diagram, and select <b>Show properties view</b>.</li> <li>2. From the Properties view, select the <b>Attributes</b> tab.</li> <li>3. Select <b>Data Source</b>, and bind to a data source.</li> </ol>

Severity and message	Cause	Action
Error: Invalid attribute name: '<Attr Name>'. A valid identifier name must begin with an alphabetic character or an underscore, followed by alphanumeric characters or underscores.	The attribute name is invalid.	Enter a valid name.
Error: Invalid attribute name: '<Attr Name>', name cannot be a Java or C# language keyword.	The attribute name is a Java or C# keyword, such as int, switch, and so on.	Do not use Java or C# keywords as attribute names.
Error: Invalid attribute name: '<Attr Name>', name cannot be a reserved attribute name.	The reserved keyword pending, replayPending, replayCounter, disableSubmit, pendingChange, lastModified, or deleted is used as an attribute name.	Enter a valid attribute name.
Error: Invalid attribute name: '<Attr Name>', the length must be less than 64.	The attribute name exceeds the 64 character maximum length.	Shorten the attribute name.
Error: Duplicate attribute name: '<Attr Name>'. The name is case-insensitive.	An attribute with the same name already exists.	Enter a valid name.
Error: Duplicate name: '<Attr Name>' in the mobile business object.	The attribute name is the same as the MBO name to which it belongs.	Enter a valid name.
Error: Duplicate name: '<Attr Name>' in the object query.	The attribute name is the same as an object query name for a given MBO.	Enter a valid name.
Error: The name: '<Attr Name>' duplicates with an operation name.	The attribute name is the same as an operation name for a given MBO.	Enter a valid name.
Error: Invalid attribute name: '<Attr Name>', name cannot be a reserved word.	A reserved keyword is used as an attribute name: deleted, lastAccessed, lastModified, isSubscribed, pending or hashCode.	Enter a valid name.

## Validation Rules

Severity and message	Cause	Action
Warning: Attribute '<Attr Name>' has no mapped data source field and no propagated load argument.	An attribute is not mapped to a column or to a load argument. This warning is generated if neither is mapped.	In the Properties view, set either the column mapping or load argument mapping. To map a load argument, set <b>Propagate to Attribute</b> on the Load arguments tab.
Warning: Data type of the column cannot map to the attribute: '<Attr Name>'.	Datatype mismatch.	Change the attribute datatype or column datatype (if it is not read-only) so they match.
Warning: Attribute '<Attr Name>' should be nullable when it's mapped to a nullable column.	The attribute is not nullable, but the column to which it maps is. The warning indicates that a failure might occur when loading or synchronizing data based on this attribute.	Change the null-ability of the attribute or the column to which it maps so they match.
Warning: Attribute data length is shorter than the mapped column length, value truncation might occur: '<Attr Name>'.	The datatype length of the attribute is shorter than the back-end column length. The value might be truncated when loading data from the back-end datasource to the cache database during refresh.	Change the attribute's length to match that of the column to which it maps.
Warning: Attribute binary data length is shorter than the mapped column length, NULL will be returned: '<Attr Name>'.	The datatype length of the binary type attribute is shorter than the back end column length, so the wrong NULL value could be loaded from the back end datasource to the cache database during refresh.	Change the attribute's length to match that of the column to which it maps.

Severity and message	Cause	Action
<p>Error: Attribute '&lt;Attr Name&gt;' of type {0}, with length greater than {2} bytes should not be used as a primary key.</p>	<p>The attribute is set as a primary key, but its length is longer than the primary key's length allowed by the server:</p> <ul style="list-style-type: none"> <li>• {0} may be STRING or BINARY</li> <li>• {2} 536870912(2G/4) for STRING</li> <li>• 0x7fffffff (Integer.MAX_VALUE, 2G-1) for BINARY</li> </ul>	<p>Change the primary key attribute's length to a valid value as indicated by the error message.</p>
<p>Error: Attribute '&lt;Attr Name&gt;' of type {0}, does not allow length greater than {2}.</p>	<p>The attribute is longer than the allowed maximum length:</p> <ul style="list-style-type: none"> <li>• {0} may be STRING or BINARY</li> <li>• {2} 536870912(2G/4) for STRING</li> <li>• 0x7fffffff (Integer.MAX_VALUE, 2G-1) for BINARY</li> </ul>	<p>Change the attribute length to a valid value as indicated by the error message.</p>
<p>Error: Column '&lt;Col Name&gt;' of type {0}, does not allow length greater than {2}.</p>	<p>The column is longer than the allowed maximum length:</p> <ul style="list-style-type: none"> <li>• {0} may be STRING or BINARY</li> <li>• {2} 536870912(2G/4) for STRING</li> <li>• 0x7fffffff (Integer.MAX_VALUE, 2G-1) for BINARY</li> </ul>	<p>Change the attribute length to a valid value as indicated by the error message.</p>

## Validation Rules

Severity and message	Cause	Action
Warning: The relationship between source mobile business object '<Source MBO Name>' and target mobile business object '<Target MBO Name>' may not work, because the load argument of target mobile business object is mapped to both attribute of source mobile business object and synchronization parameter of target mobile business object.	If a load argument is bound to both a foreign attributes and a synchronization parameter, then the relationship between the source and target MBOs may not work.	Select the correct mapping for load arguments according to a valid scenario.
Error: All load argument(s) of target mobile business object '<MBO Name>' must be filled either from attribute(s) of source mobile business object(s) or from synchronization parameter(s) of target mobile business object, a combination is not allowed.	If an MBO has multiple load arguments, then they can all come from sync parameters, or all from "foreign attributes" (for example, attributes of a parent MBO), but a mixture is not supported. Otherwise, deployment will fail.	Ensure that load arguments of MBOs come from sync parameters, or that all come from "foreign attributes".
Error: Mobile business object "<MBO Name>" with online cache policy must have at least one load argument with "Propagate To" attribute set.	Online MBO does not have a load argument with "Propagate To" set.	Assign a "Propagate To" attribute for at least one load argument.
Error: Invalid < object type > name: name cannot be empty.	The <object type> name cannot be empty.	Enter a valid value for the name of the <object type>.
Error: Primary key attribute "<Attribute name>" cannot have data type "<BigString/BigBinary>".	Set a BigString/BigBinary attribute as primary key.	Do not set the attribute as primary key who is BigString/BigBinary.
Warning: The length of column "<Column name>" is unknown, so the length of attribute "<Attribute name>" is set to <default length> by default. Update the attribute length to avoid truncation if necessary.	If the length is not specified, set a default value.	Update the value by hand.

Severity and message	Cause	Action
Error: The attribute name "<ATTRIBUTE-UTENAME>" is not valid. The name must not consist of a prefix "old" followed by another attribute name of this MBO.	Change the attribute name.	An attribute name of old $XXX$ , where $XXX$ is an existing attribute name is not allowed and must be changed.
Warning: The mobile business object "<MBONAME>" should have an attribute defined as a primary key.	Due to this error, packages with MBO's that do not have a primary key cannot be deployed.	To achieve optimal performance of insert, update, and delete operations, as well as client synchronization, the MBO definition must include a well defined primary key. Local MBOs do not require a primary key definition.

## Parameter and Argument Validation Rules and Error Messages

These validation rules and error messages apply to mobile business object parameters and arguments, and are enforced during MBO development, and, in some cases, when you deploy the MBO to Unwired Server. Messages are generated when validation rules are violated.

**Table 8. Parameter and argument validation rules and error messages**

Severity and message	Cause	Action
Error: Invalid parameter name: '<Parameter Name>'. A valid identifier name must begin with an alphabetic character or an underscore, followed by alphanumeric characters or underscores.	The parameter name is invalid.	Enter a valid name.
Error: Invalid parameter name: '<Parameter Name>', name cannot be a Java or C# language.	The parameter name is a Java or C# keyword, such as binary, boolean, byte, char, date, datetime, time, decimal, double, float, object, int, integer, long, short, string, switch, and so on.	Do not use Java or C# keywords as parameter names.

## Validation Rules

Severity and message	Cause	Action
Error: Invalid parameter name: '<Para Name>', the length must be less than 100.	The length of the parameter name is longer than 100 characters.	Shorten the parameter name to less than 100 characters.
Error: The argument '<Arg Name>' in operation '<Oper Name>' is not nullable and does not have a parameter mapped, thus it must have a default value.	A default value is not specified for the operation's parameter, which may result in a runtime operation failure.	Specify a default value for the argument.
Error: Attribute '<Attr Name>' cannot be mapped to more than one parameter.	An attribute has more than one <b>Propagate to</b> mapping.	Remove additional mappings.
Error: The data type of the parameter '<Para Name>' can not map to the data type of the argument.	The datatype of the parameter and the datatype of the argument do not match.	Change the datatype of the parameter to match that of the argument. Or change the type of the argument if it is not read-only. <hr/> <b>Note:</b> Make sure the argument type matches that of the data source for JDBC data sources.
Warning: Parameter '<Para Name>' should not be nullable when it is mapped to a non-nullable argument.	The nullability of the parameter does not match that of the argument. For a non-nullable argument, the mapping parameter should also be non-nullable.	Change the parameter to non-nullable.
Error: Data type of the bound personalization key cannot map to the parameter: '<Para Name>'.	Datatype mismatch.	Change the datatype of the parameter or the datatype of the personalization key so they match.



Severity and message	Cause	Action
Warning: Parameter data length is shorter than the bound personalization key's data length, value truncation might occur: '<Para Name>'	The length of the datatypes do not match.	Change the length of the parameter or that of the personalization key so they match.
Warning: A non-nullable parameter '<Para Name>' cannot map to the nullable personalization key: '<PK Name>'.	A non- nullable parameter is mapped to a nullable personalization key.	Change the nullability of the parameter or that of the personalization key.
Warning: The maximum length of the operation parameter is different from the fill from attribute's: '<Para Name>'.	The datatype lengths do not match. At runtime, when the attribute passes its value to the parameter, the value may be truncated.	Change the datatype length so they match.
Error: Binary parameter default value cannot exceed 16384 byte: '<Para Name>'.	The default value is too long.	Shorten the default value.
Error: Synchronization parameter '<Sync Para Name>' cannot be mapped to more than one argument.	Two or more arguments map to the same synchronization parameter.	Change the mapping so only one argument maps to the synchronization parameter.
Error: Data type of the selected old value argument cannot map to the parameter '<Para Name>' of operation '<Oper Name>'.	Datatype mismatch.	Change the datatype of the <b>Old value argument</b> or that of the parameter so they match.

## Validation Rules

Severity and message	Cause	Action
Error: Argument '<Arg Name>' of type <Type>, does not allow length greater than <Arg Length>.	The argument length exceeds the maximum length allowed for its datatype.	Shorten the argument length to a valid length for the given type according to the message.
Error: Invalid default value '<Default Value>' for the '<Type>' data type: '<Arg Name>'.	The argument default value is not valid for its datatype.	Change the default value to conform to the argument's datatype.
Warning: Default value exceeds maximum length: '<Arg Length>'.	The argument default value length exceeds the argument's length.	Change the default value to conform to the argument's length.
Warning: Non-nullable argument '<Arg Name>' cannot map to a nullable personalization key: '<Per Key Name>'.	A non-nullable argument is mapped to a nullable personalization key.	Change the nullability of the argument or the personalization key so they match.
Error: Invalid <object type> name: name cannot be empty.	The <object type> name cannot be empty.	Enter a valid value for the name of the <object type>.
Warning: Personalization key "{0}" with storage type "{1}" assigned to load argument: "{2}" will not be used because the mobile business object has a DCN cache policy.	Do not use transient personalization keys assigned to load arguments if the MBO uses a DCN cache group policy.	Remove the Personalization Key mapping for the argument from the load arguments tab in the MBO Property view.

Severity and message	Cause	Action
<p>Error: If client parameter's data type is bigString/bigBinary or a structure type with bigString/bigBinary attributes, show as an error like "Structure " &lt; Structure name &gt; " contains field " &lt; attribute name &gt; " with data type " &lt; BigString/BigBinary &gt; ", and cannot be used as a client parameter " &lt; client parameter name &gt; ."</p>	<p>An error message displays if you generate a client parameter and use a structure that contains BigString/BigBinary field(s).</p>	<p>To provide client parameters for BigString/BigBinary datatypes, or a structure with BigString/BigBinary fields, use compatible types: String for BigString, Binary for BigBinary. For a structure, copy the original structure in Workspace Navigator, and change the large object type fields to String/Binary, then use the new structure as the datatype of the client parameter.</p>
<p>Error: Personalization key can't be directly assigned to load argument: '&lt;Arg Name&gt;' of mobile business object which has a scheduled cache policy.</p>	<p>Do not use reserved parameters as an attribute name: count_bad_playback, error_category,error_message, last_good_playback, last_modified, last_playback, uagid, uapbk and uask.</p>	<p>Change the attribute name.</p>
<p>Warning: Invalid parameter name: '&lt;Para Name&gt;' name cannot be a reserved parameter name.</p>	<p>Bound to a wrong personalization key.</p>	<p>Redefine the personalization key or remove the mapping.</p>
<p>Warning: Argument '&lt;Arg Name&gt;' should not be bound to personalization key '&lt;PK Name&gt;' whose default value is set to '&lt;value&gt;' when its 'Propagate to' attribute '&lt;Attr Name&gt;' is a primary key.</p>	<p>A nullable personalization key is mapped to a non-nullable parameter.</p>	<p>Change the nullability of the parameter or that of the personalization key so they match.</p>

## Validation Rules

Severity and message	Cause	Action
Warning: Non-nullable parameter "<Parameter name>" cannot map to the nullable personalization key: "<PK name>".	The length of datatypes do not match. When the parameter passes the value to the argument, the value might be truncated.	Change the datatype length so they match.
Warning: Argument data length is shorter than the 'Parameter' length, value truncation might occur: '<Para Name>'.	The length of datatypes do not match. When the argument value is passed to the propagated attribute, the value might be truncated.	Change the datatype length so they match.
Warning: Argument data length is larger than that of the attribute it propagates to, value truncation might occur: '<Argument Name>'.	The datatype length of the operation argument is shorter than its 'fill from attribute' datatype length, and the value might be truncated.	Change the datatype length so they match.
Warning: Argument data length is shorter than the <length> length, value truncation might occur: "<Operation name>".	Only 'UPDATE' type operations support old value arguments.	Check the definition of the operation, this might be an 'UPDATE' type operation.
Error: Operation '<Oper Name>' does not support old value argument.	An argument can not be used both as an argument and as an 'Old Value Argument'.	Change mapping of argument or the 'Old Value Argument'.
Warning: 'Old Value Argument' and 'Argument' cannot be the for the operation '<Oper Name>'.	Need to specify the 'Fill from attribute', before the 'old Value Argument' can take effect.	Specify the 'Fill from Attribute'.

Severity and message	Cause	Action
Warning: The old value argument '<Arg Name>' of operation '<Oper Name>' will be ignored when the argument '<Arg Name>' has no 'Fill from attribute'.	The client parameter of the operation has no mapping argument.	Specify the argument to map to the client parameter.
Warning: No argument is mapped to client parameter '<Para Name>'.	The nullability between personalization key and parameter is inconsistent.	Unify the nullability between personalization key and parameter.
Error: The nullability of the bound personalization key '<personalization key name>' is not consistent with parameter '<parameter name>'.	Null is set as the default value for a non-nullable argument.	Provide a valid default value for the argument.
Error: The default value of a non-nullable argument should not be set to "<NULL>": "<Arg Name>".	Set null as default value for a non-nullable synchronization parameter.	Provide a valid default value for the synchronization parameter.
Warning: The default value of a non-nullable synchronization parameter should not be set to "<NULL>": "<Arg Name>".	The mapped-to attribute of a parameter has string/binary type but does not have a length set, which may cause the query to fail since BLOB/CLOB filtering is currently unsupported.	Explicitly set the length for the attribute with type of string/binary.
Warning: The mapped attribute of parameter "<parameter name>" does not have length set which may cause the query to fail.	The mapped-to attribute of a parameter is a string/binary datatype, but does not have the length set, which may cause the query to fail since Unwired Platform currently does not support filtering of BLOB/CLOB.	Explicitly set the length for the attribute with datatype of string/binary.

## Validation Rules

Severity and message	Cause	Action
Error: The mapped attribute of parameter "<Synchronization parameter name>" does not have length set which may cause the synchronization to fail.	Synchronization parameter maps to a BigString/BigBinary attribute.	Map the synchronization parameter to a non BigString/BigBinary attribute.
Warning: Synchronization parameter "<Parameter name>" should not be mapped to attribute "<Attribute name>" with data type "<BigString/BigBinary>".	Set a default value if the length is not specified.	Manually update the value.
Error: The length of parameter "<Parameter name>" is not specified, set to <Default length> by default. Update the length to avoid truncation if necessary.	Datatypes do not match.	Change the datatype of the attribute so they match.
Warning: Data type of the attribute cannot map to the argument: "<Arg name>".	The nullability setting of the datatypes do not match.	Change datatype nullability so they match.
Error: Argument "<Arg name>" should not be nullable when it is not used in relationship and its "propagate to" attribute is not nullable.	Wrong propagate-to attribute setting.	Unset the propagate-to attribute.
Warning: The attribute "<Attr Name>" has been mapped in relationship "<Relationship Name>". It can not be set as propagate-to attribute.	The nullability doesn't match.	Change nullability to let them match.

Severity and message	Cause	Action
Error: The nullability of the argument '<Arg Name>' should be the same as the attribute it is filled from.	Two or more client parameters map to the same argument.	Change the mapping; only allow one parameter to map to the argument.
Warning: Client parameter '<Para Name>' cannot be mapped to more than one argument.	The length of datatypes does not match. When personalization key pass value to the argument, value might be truncated.	Change the datatype length so they match.
Warning: Argument data length is shorter than the bound personalization key's data length, value truncation might occur: '<Arg Name>'.	A nullable personalization key is mapped to a non-nullable argument.	Change the nullability attribute of the personalization key.
Warning: Non-nullable argument "<Argument name>" cannot map to a nullable personalization key: "<PK name>".	Invalid Personalization key mapping.	Remove the mapping.
Error: Personalization key '<PK Name>' with storage type "<Storage Type>" assigned to load argument: '<Arg Name>' will not be used because the mobile business object has a DCN cache policy.	Java package names must follow Java naming conventions.	Do not use spaces in the name.
Error: A package name must not start or end with a blank.	User specifies name with unusable characters like %!\$%&/()=,. Even though it does not follow Java naming convention, package names can start with an uppercase letter.	Do not use those characters in the name.

## Operation Validation Rules and Error Messages

Validation rules are enforced during MBO development, and, in some cases, when you deploy the MBO to Unwired Server. Messages are generated when validation rules are violated.

**Table 9. MBO operation validation rules and error messages**

Severity and message	Cause	Action
Error: Invalid operation name: '<Oper Name>'. A valid identifier name must begin with an alphabetic character or an underscore, followed by alphanumeric characters or underscores.	The operation name is invalid.	Enter a valid name.
Error: Invalid operation: '<Oper Name>', name cannot be a Java or C# language keyword.	The operation name is a Java or C# keyword, such as binary, boolean, byte, char, date, datetime, time, decimal, double, float, object, int, integer, long, short, string, switch, and so on.	Do not use Java or C# keywords as operation names.  <b>Note:</b> "create", "update", and "delete" are reserved keywords for operations if used for a different type of operation than that of the name. For example, an operation named "create" can only be used for a Create operation, and not for Update or Delete operations.
Error: Invalid operation name: '<Oper Name>', the length must be less than 64.	The length of the operation name is longer than 64 characters.	Enter a valid name shorter than 64 characters.
Error: Duplicate operation name: '{0}'. The name is case-insensitive.	Duplicate operation name.	Enter a valid and unique name.
Error: Duplicate name: '<Oper Name>' in the mobile business object.	The operation name is the same as the MBO name to which it belongs.	Enter a valid name.
Error: Duplicate name: '<Oper Name>' in the object query.	The operation name is the same as an object query for this MBO.	Enter a valid name.
Error: The name: '<Oper Name>' is a duplicate of the query name "FindAll".	User checks to generate 'FindAll' query and also define an operation named 'FindAll'.	Use another valid name or uncheck the button to generate 'FindAll' query.



Severity and message	Cause	Action
Error: The name: '<Oper Name>' duplicates with an attribute name.	The operation name is the same as an attribute name for the MBO.	Enter a valid name.
Warning: Operation name '<Oper Name>' starts with 'findby', which may cause name conflict when generating client code.	The operation name cannot start with "findby."	Enter a valid name.
Error:Invalid '<Oper Name>' name: name cannot be empty.	User does not specify the name for operation.	Specify a valid name for operation.
Warning: Operation is not bound to the back-end data source: '<Oper Name>'.	The operation is not bound to a data source.	Bind a data source to the operation.
Error: If a mobile business object has no attributes, the operation '<Oper Name>'s type must be set to 'OTHER'.	An operation type other than "other" has been defined for the MBO, and no attributes are defined.	Only "other" operations support no attributes. Change the operation type to "other", or add attributes to the MBO.
Error: argument '<Para Name>' cannot have 'Fill from attribute' as the type of the operation '<Oper Name>' is 'OTHER'.	An "other" operation argument has a <b>Fill from attribute</b> mapping set.	Since "other" operations are static from a code generation perspective, remove all <b>Fill from attribute</b> mappings from "other" operation arguments.
Error: The response representation of the "UPDATE" operation '<Operation Name>' with "Apply results to the cache" cache policy does not have an XSLT defined.	For restful Web service data sources, If an MBO update operation does not have an XSLT associated with it and the "Apply results to the cache" cache policy is set, after deployment to the server, NullPointerException will be thrown during the execution.	Add an XSLT for the response representation.

## Validation Rules

Severity and message	Cause	Action
Error: The "UPDATE" operation '<Operation Name>' with "Apply results to the cache" cache policy does not have response representation.	For restful web service data source, If an update operation of MBO doesn't have an XSLT associated with it and the "Apply results to the cache" cache policy is set, after deployment to the server, NullPointerException will be thrown during the course of execution.	Add a response representation for the Update operation and bind an XSLT for it.
Error: The "UPDATE" operation "<Operation Name>" with "Apply results to the cache" cache policy does not have XSLT.	For SOAP or WSDL based data sources, If an MBO update operation does not have an XSLT associated with it and the "Apply results to the cache" cache policy is set, after deployment to the server, NullPointerException will be thrown during execution.	Add an XSLT for update operation.
Error: Operation "{0}" should not have the "Invalidate the cache" policy since the MBO is in a "DCN" cache group.	The 'Invalidate the cache' policy was set on an operation whose MBO is in a 'DCN' cache group.	Remove the policy from the operation.
Warning: No arguments are available from the root element: "{0}"	A root element was specified without an argument from the "Define Representation" dialog for a RESTful Web service Request.	Select a root element that includes at least one argument.
Warning: No arguments are available from the root element "{0}" of the request representation "{1}"	The MBO developer clicks "OK" to close the "Define Representation" dialog after specifying a root element without an argument when creating a new operation for a RESTful MBO.	Click "Edit" again to invoke the Define Representation dialog and select a valid root element.

Severity and message	Cause	Action
Error: The default XSLT cannot be generated. Define the XSLT manually.	The MBO developer selects a root element without an argument from the "Define XSLT" dialog when adding an XSLT to the response.	Click "Cancel" to close the "Define XSLT" dialog, select a valid root element, and click "Add" again.
Warning: Specify an XML representation for the successful state transfer	An XSLT has not been added from the "Define Representation" dialog for a RESTful response.	Click "Add" to add an XSLT.
Error: The details or reference of the representation "{0}" is missing in operation: "{1}"	For a RESTful response, "Response" is selected but an XSLT has not been added, or a reference set. For RESTful request, "Request" is selected, but an XSD file has not been set, or a valid root element is missing. Set from the "New Operation" wizard for a RESTful Response.	If the error is for response, then click "Edit" button for response to invoke "Define Representation" dialog, and add at least one XSLT. If the error is for request, then click "Edit" button for request to invoke "Define Representation" dialog select a valid root element .
Warning: For operation "<name>" with cache policy "Apply merge of operation input/output", if any output mappings are defined, all primary key attributes should be mapped to output results. The following primary key attributes are not mapped: <list of attribute names>.	The operation is using the "Apply merge of operation input/output" cache policy, and at least one output mapping was added for an attribute of the MBO. There may be output mappings for some primary key attributes (if there is more than one primary key attribute defined), but not all primary key attributes have output mappings defined.	Add output mappings for all primary key attributes of the MBO that has the operation.
Error: For operation "<name>" with cache policy "Apply output of ENTITY READ", all primary key attributes must be mapped to output results. The following primary key attributes are not mapped: <list of attribute names>.	The operation is using the "Apply output of ENTITY READ operation" cache policy, and not all primary key attributes have output mappings defined.	Add output mappings for all primary key attributes of the MBO that has the operation.

## Validation Rules

Severity and message	Cause	Action
<p>Error: For operation "&lt;name&gt;" with cache policy "Apply output of ENTITY READ", all primary key attributes must be mapped to output results, but the operation does not have output.</p>	<p>The operation is using the "Apply output of ENTITY READ operation" cache policy, which requires all primary key attributes be mapped to operation results. But the operation does not define any output results.</p>	<p>Define output results for the operation that can be mapped into all the primary key attributes of the MBO containing the operation.</p>
<p>Warning: Operation "&lt;name&gt;" does not have output mappings to all MBO attributes in the composite graph and will result in inconsistent data contents between Unwired Server Cache and EIS data source.</p>	<p>The operation is using the "Apply merge of operation input/output" cache policy, but does not have output mappings defined for all attributes of all MBOs in the composite graph. As a result, the cache may only reflect changes to the attributes that have output mappings, but not the other attributes of the composite graph.</p>	<p>Define output mappings for all attributes of all MBOs in the composite graph.</p>
<p>Error: Operation "&lt;name&gt;" should not set both "Invalidate the cache" and "Apply merge" policies when multiple output results are mapped.</p>	<p>If the operation uses the "Apply merge of operation input/output" cache policy and outputs to all the MBOs in the composite graph, the "Invalidate the cache" policy is redundant and has a negative impact on performance.</p>	<p>Select either the "Invalidate the cache" or "Apply merge of operation input/output" cache policy, but not both.</p>
<p>Error: A mobile business object cannot have more than one operation of type "ENTITY READ". Mobile business object "&lt;name&gt;" already has an "ENTITY READ" operation.</p>	<p>There is already an ENTITY READ operation defined for the MBO.</p>	<p>Do not define more than one ENTITY READ operation in a composite graph.</p>

Severity and message	Cause	Action
Error: All primary key attributes, except those propagated from load arguments, must be mapped into input arguments for the "ENTITY READ" operation "<name>".	Some or all of the MBO's primary key attributes are not used as Fill from attributes for the operation input arguments.	Use all primary key attributes of the MBO containing the ENTITY READ operation as Fill from attributes for the operation arguments (primary key attributes propagated from load arguments do not need to be mapped).
Error: An "ENTITY READ" operation is defined for mobile business object "<mbo name>" but can only be defined for the root mobile business object of the composite graph: "<root mbo name>".	An ENTITY READ operation is used to update the cache with an instance of a composite graph; this is currently supported only for the root of the composite graph.	Do not define an ENTITY READ operation on child MBOs of a composite graph; only on the root MBO.
Error: The operation "<name>" has set the "Apply output of ENTITY READ" cache policy, but there is no "ENTITY READ" operation defined.	The "Apply output of ENTITY READ operation" cache policy is used to invoke an ENTITY READ operation to update the cache, after the original operation is done. It is an error if the ENTITY READ operation is not defined.	Either do not use the "Apply output of ENTITY READ operation" cache policy, or define an ENTITY READ operation.
Error: An operation in a mobile business object that is the target of a composite relationship should not set the "Apply output of ENTITY READ" cache policy: "<name>".	The "Apply output of ENTITY READ operation" cache policy must only be used in an operation defined on the root MBO of a composite graph.	Do not use the "Apply output of ENTITY READ operation" cache policy in an operation in a child MBO.
Error: All attributes, except those propagated from load arguments, must be mapped to results for the "ENTITY READ" operation "<name>".	An ENTITY READ operation updates the cache with an instance of a complete composite graph, and requires all attributes of the composite graph have output mappings defined.	Make sure output results are defined that can be used as input to all attributes of all MBOs in the composite graph, and then define output mappings for all attributes of all MBOs in the composite graph (attributes propagated from load arguments do not require output mappings).

## Validation Rules

Severity and message	Cause	Action
<p>Error: For operation "&lt;name&gt;" with cache policy "Apply merge of operation input/output", when mapping multiple results, all attributes must be mapped (except attributes propagated from load arguments).</p>	<p>For a Create or Update operation using the "Apply merge of operation input/output" cache policy to update the cache with a complete instance of a composite graph, output mappings must be defined for all attributes of all MBOs in the composite graph. This error displays if attributes from more than one MBO have output mappings defined, but not all attributes from all MBOs.</p>	<p>Define output mappings for all attributes of all MBOs in the composite graph (attributes propagated from load arguments do not require output mappings).</p>
<p>Error: There is no output result for the "ENTITY READ" operation "&lt;name&gt;".</p>	<p>An ENTITY READ operation updates the cache with an instance of a complete composite graph, and requires all attributes of the composite graph have output mappings defined.</p>	<p>Make sure output results are defined that can be used as input to all attributes of all MBOs in the composite graph.</p>
<p>Warning: "ENTITY READ" operation "&lt;name&gt;" has been defined, but no operation has set the "Apply output of ENTITY READ" cache policy.</p>	<p>If no operation sets the "Apply output of ENTITY READ operation" cache policy, the ENTITY READ operation is not used.</p>	<p>Define a Create or Update operation on the root MBO of the composite graph, which uses the "Apply output of ENTITY READ operation" cache policy.</p>
<p>Warning: "CREATE" Operation "&lt;name&gt;" uses the "Apply output of ENTITY READ" cache policy, so only primary key attributes must be mapped to output columns. Additional output mappings will be ignored.</p>	<p>The ENTITY READ operation invoked by the CREATE operation only requires the primary key as input, so the CREATE operation only needs to output the primary key.</p>	<p>Remove output mappings for non-primary key attributes.</p>

Severity and message	Cause	Action
Warning: "UPDATE" Operation "{0}" uses the "Apply output of ENTITY READ" cache policy, so output mappings are not required and	The ENTITY READ operation invoked by the UPDATE operation only requires the primary key as input, which should be provided as input to the UPDATE operation, and so the UPDATE operation does not need to define any output mappings.	Remove all output mappings.
Warning: Operation "<name>" defines output mappings but does not set an "Immediately update the cache" policy. Output mappings will be ignored.	Output mappings are only needed if the operation updates the cache directly or via an ENTITY READ operation.	Remove all output mappings, or set an "Immediately update the cache" policy.

## Relationship Validation Rules and Error Messages

Validation rules are enforced when defining a relationship between two MBOs and, in some cases, when you deploy the MBO to Unwired Server. Messages are generated when validation rules are violated.

**Table 10. Relationship validation rules and error messages**

Severity and message	Cause	Action
Error: Duplicate relationship name: '<Relationship Name>' case insensitive.	A duplicate relationship name (source attribute name) is defined.	Change the auto-generated name of the source attribute in the Relationship Creation wizard, Edit dialog, or Properties view.
Error: Relationship source attribute name <Relationship Name> is a duplicate of the existing source attribute name '<Attribute name>'.	The source attribute's name of the current relationship was duplicated with a source attribute's name of another relationship. The two relationship have same source object (Source MBO).	Either change the source attribute's name for the current relationship or change the duplicated source attribute's name in the other relationship.
Error: No mapping information specified for the relationship: '<Relationship Name>'.	The relationship does not have a mapped attribute or mapped parameter.	Map the attribute or parameter in the Relationship Creation wizard, Edit dialog, or Properties view.

## Validation Rules

Severity and message	Cause	Action
Error: Relationship name '<Relationship name>' is a duplicate of the existing attribute name '<Attribute name>'.	The relationship's name is a duplicate of the source MBO attribute's name.	Change the attribute name in the source MBO, or change the source attribute's relationship name.
Error: Relationship target attribute name '<Attribute name>' is a duplicate of the existing attribute name '<Attribute name>'.	The relationship's target attribute name is a duplicate of an attribute name in the target MBO.	Change the target MBO attribute name, or change the target attribute's name.
Error: Relationship target attribute name '<Attribute name>' is a duplicate of the existing source attribute name '<Attribute name>'.	The relationship's target attribute name is a duplicate of the source attribute's name in another relationship.	Change this relationship's target attribute name, or change the duplicated source attribute name.
Error: Attributes must be defined for '<MBO Name>' to create a relationship.	No attributes are defined in the source MBO or the target MBO.	Create attributes in the source or target MBO.
Error: Primary key attributes must be defined for '<MBO Name>' in the relationship '<Relationship name>'.	Primary key attributes are not defined in the source MBO or the target MBO.	<p>Set the primary key attribute in the source MBO in one-to-many bidirectional, unidirectional, and one-to-one bi-directional relationships.</p> <p>Set the primary key attribute in the target object (MBO) in many-to-one bidirectional, unidirectional, and one-to-one unidirectional relationships. You can either:</p> <ul style="list-style-type: none"> <li>• Set the related attribute as primary key on source or target MBOs according to the relationship's type.</li> <li>• automatically set or unset the mapped attribute as a primary key preference setting (from the Preferences menu, select <b>Sybase, Inc &gt; Mobile Development &gt; Miscellaneous &gt; Relationship</b>).</li> <li>• Map the attribute or parameter between the source and target MBO.</li> </ul>



Severity and message	Cause	Action
Error: The primary key attribute '<Attribute name>' must be mapped in the relationship '<Relationship name>'.	There is no primary key mapping for the source or target MBO attribute.	Unset the attribute as a non-primary key, or map the attribute in the relationship. For one-to-many bidirectional, unidirectional and one-to-one bidirectional relationships, map all primary key attributes of the source MBO. For many-to-one bidirectional, unidirectional and one-to-one unidirectional relationships, map all primary key attributes of the target MBO.
Error: The non-primary key attribute '<Attr Name>' should not be mapped in the relationship '<Relationship Name>'.	At least one non-primary key attribute is mapped in the relationship.	Set the mapped attribute as a primary key, or unmap the attribute in the relationship. For one-to-many bidirectional, unidirectional and one-to-one bidirectional relationships, unmap all non-primary key attributes of the source MBO. For many-to-one bidirectional, unidirectional and one-to-one unidirectional relationships, unmap all non-primary key attributes of the target MBO.
Error: The data type of attribute 'Attribute name' does not match 'Attribute name' in the relationship.	The datatype of the mapped attribute/parameter does not match.	Map the attribute/parameter to a target with the same datatype.
Error: Cannot create relationship between different type of mobile business object.	A relationship between different types of MBOs exists.	Create relationships only between the same types of MBOs.
Error: Cannot create relationship between structures.	A relationship between structures exists.	Relationships between structures is not currently supported.
Error: The target mobile business object '<MBO Name>' can not be used in the composite relationship 'relationship name' since it already was used in a composite relationship as a target.	One MBO is used in multiple composite relationships as a target MBO.	In composite relationships, define only one MBO as a target MBO.

## Validation Rules

Severity and message	Cause	Action
Error: The argument '<Argument name>' should set the propagate-to attribute in the relationship between '<MBO Name>' and '<MBO Name>'.	The mapped load argument does not have a propagate-to attribute in the relationship.	Unmap the load argument, or set the propagate-to attribute for the load argument.
Error: As the associated propagate-to attribute is a primary key, the argument '<Argument name>' should be mapped to an attribute or a argument in the relationship with '<MBO Name>'.	The load argument, which has a propagate-to attribute primary key, is not mapped in the relationship.	Unset the propagate-to attribute as non-primary key. Or,  Map the load argument in the relationship for both the target and source MBO: for one-to-many bidirectional, unidirectional and one-to-one bidirectional relationship, map all load arguments where the propagate-to attribute are primary key attributes of the source MBO. For many-to-one bidirectional, unidirectional and one-to-one unidirectional relationship, map all load arguments where the propagate-to attribute are primary key attributes of the target MBO.
Error: The argument '<Argument name>' was mapped in the relationship '<relationship name>'. Its propagate-to attribute '<Attribute name>' must be a primary key.	The load argument's propagate-to attribute is mapped in the relationship, but is not a primary key.	Set the propagate-to attribute as a primary key, or Unmap the load argument in the relationship: <ul style="list-style-type: none"> <li>• Unmap all propagate-to attribute load arguments that are non-primary key attributes in the source MBO if used in one-to-many bidirectional, unidirectional, and one-to-one bidirectional relationships.</li> <li>• Unmap all propagate-to attribute load arguments that are non-primary key attributes in the target object (MBO) if used in many-to-one bidirectional, unidirectional, and one-to-one unidirectional relationships.</li> </ul>

Severity and message	Cause	Action
<p>Error: The primary key attribute '&lt;Attribute name&gt;' was not used in the relationship '&lt;Relationship name&gt;'.</p>	<p>The primary key attribute is not mapped in the relationship, but is mapped in another relationship.</p>	<p>Map the attribute in the relationship:</p> <ul style="list-style-type: none"> <li>• Map all primary key attributes of the source MBO in one-to-many bidirectional, unidirectional, and one-to-one bidirectional relationships.</li> <li>• Map all primary key attributes of the target MBO in many-to-one bidirectional, unidirectional, and one-to-one unidirectional relationships.</li> </ul>
<p>Error: The propagate-to attribute of the argument '&lt;Argument name&gt;' is a primary key and must be used in the relationship '&lt;Relationship name&gt;'.</p>	<p>The propagate-to attribute load argument, which is a primary key, is not mapped in the current relationship, but is mapped in another relationship.</p>	<p>Map the load argument in the relationship:</p> <ul style="list-style-type: none"> <li>• Map all propagate-to attribute load arguments that are primary key attributes of the source MBO in one-to-many bidirectional, unidirectional, and one-to-one bidirectional relationships.</li> <li>• Map all propagate-to attribute load arguments that are primary key attributes of the target MBO in many-to-one bidirectional, unidirectional, and one-to-one unidirectional relationships.</li> </ul>
<p>Error: Relationship from 'MBO Name' to '&lt;MBO Name&gt;' is invalid, as it causes a circular relationship.</p>	<p>The relationship is circular.</p>	<p>Remove the relationship. Circular relationships are not allowed.</p>
<p>Warning: Relationship from '&lt;MBO Name&gt;' to '&lt;MBO Name&gt;' references two mobile business objects in different synchronization groups. Mobile business objects with relationship should be grouped in the same synchronization group.</p>	<p>The two MBOs in the relationship do not belong to the same synchronization group.</p>	<p>Move the source and target MBOs into the same synchronization group.</p>

## Validation Rules

Severity and message	Cause	Action
<p>Error: Composite relationship from '&lt;MBO Name&gt;' to '&lt;MBO Name&gt;' references two mobile business objects in different synchronization groups. Mobile business objects with composite relationship should be grouped in the same synchronization group.</p>	<p>The two MBOs in a composite relationship are not in the same synchronization group.</p>	<p>Move the source and target MBOs into a same synchronization group.</p>
<p>Error: Relationship '&lt;relationship name&gt;' references two mobile business objects in different cache groups. Mobile business objects with relationship should be put in the same cache group.</p>	<p>The two MBOs in the relationship do not belong to the same cache group.</p>	<p>Move the source and target MBOs into a same cache group.</p>
<p>Error: Composite relationship "&lt;relationship_name&gt;" references two mobile business objects in different cache groups. Mobile business objects with composite relationship must be put in the same cache group.</p>	<p>The two MBOs in a composite relationship are not in the same cache group.</p>	<p>Move the source and target object (MBO) into the same cache group.</p>
<p>Error: The secondary mobile business object '&lt;MBO Name&gt;' can not be used in the relationship '&lt;Relationship name&gt;' as source object.</p>	<p>In a relationship that involves multiple MBOs created from the same EIS operation, the secondary (subsequent) MBO is the relationship's source object and the target MBO is its primary MBO, and the load argument of the primary MBO is mapped.</p>	<p>Do not map the load argument of the source MBO, or swap the MBOs so that the secondary MBO is the primary MBO.</p>
<p>Error: The secondary mobile business object '&lt;MBO Name&gt;' can not be used in the relationship '&lt;Relationship name&gt;' as target object.</p>	<p>In a relationship that involves multiple MBOs created from the same EIS operation, the secondary MBO is the target object and its load argument is mapped.</p>	<p>Do not map the load argument of the target object (MBO) unless the source MBO is also the primary MBO, or swap the MBOs so that the secondary MBO is the primary MBO.</p>

Severity and message	Cause	Action
Error: Multiple relationships between "<MBO1_name>" and "<MBO2_name>" caused circular dependencies, must set one side of the foreign key attributes to nullable.	The multiple relationships between MBOs form circular dependencies with foreign keys non-nullable in both MBO. Unwired Server cannot load MBOs for this case.	Make the foreign key attributes nullable in either source or target side.
Error: Relationship "<Relationship name>" involves mobile business objects with online cache policy must be a composite relationship.	The relationship in online cache group is not a composite relationship.	Make the relationship a composite relationship.
Error: Relationship has two mobile business objects with independent load definitions in the online cache group, and this is not allowed.	The relationship is established between non-commingled MBOs in online cache groups.	Remove the unsupported relationship.
Error: Relationship "<Relationship Name>" references mobile business objects across "<Cache group type1>" and "<Cache group type2>" cache groups.	Mobile business objects with relationship should be put in the same cache group. Relationship established on MBOs across online and non-online cache groups.	Make the source and target MBOs cache groups are consistent—either all in an online cache group, or all in a non-online cache group.
Error: Mobile business object "<MBO_Name1>" must be set as partition by requester and device identity to reference the requester-partitioned mobile business object "<MBO_Name2>".	A non requestor-partition reference a requestor-partitioned mobile business object.	Change either the source MBO setting or target MBO setting so that no such reference appears.  <b>Note:</b> It is permissible for a requestor-partitioning MBO to reference a non requestor-partitioning MBO.
Error: Relationship target attribute name '<Attribute name>' is a duplicate of the existing target attribute name '<Attribute name>'.	The target attribute's name of the current relationship is a duplicate of a target attribute's name of another relationship. The two relationships have the same target object (Target MBO)	Either change the target attribute's name for the current relationship, or change the duplicated target attribute's name in the other relationship.

## Validation Rules

<b>Severity and message</b>	<b>Cause</b>	<b>Action</b>
Error: Relationship source attribute name '<Attribute anme>' is a duplicate of the existing target attribute name '<Attribute name>'.	The source attribute's name of the current relationship is a duplicate of a target attribute's name of another relationship, which target object (target MBO) is the source object (Source MBO) of the current relationship.	Either change the source attribute's name for the current relationship, or change the duplicated target attribute's name in the other relationship.

### Relationship Guidelines and Restrictions

Follow these guidelines when configuring MBOs and enterprise information systems (EISs) used in relationships.

**Table 11. Relationship Guidelines**

Guideline	Example
<p>These examples use SQL Anywhere to illustrate how to properly configure the EIS so delete operations behave as expected</p>	<ul style="list-style-type: none"> <li>Example one consists of two MBOs:           <pre>Address: addr_id, street, city, country</pre> <pre>Employee: emp_id, name, addr_id (foreign key references Address.addr_id)</pre> <p>The MBOs have a relationship defined, for example, a composite, uni-directional, one-to-one relationship:</p> <pre>Employee(Parent) --&gt; Address(Child)</pre> <p>At runtime, if you delete Employee, Unwired Server attempts to delete the child Address first, which fails because of the Employee.addr_id-to-Address.addr_id reference. For the delete operation to perform as expected, the addr_id foreign key must be defined in the EIS database as <b>on delete set null</b> to enable deletion.</p> <p>This requirement applies regardless of the type of relationship defined between the MBOs.</p> </li> <li>Example two also uses the Employee and Address MBOs in a relationship that has mutual references:           <pre>Address: addr_id, street, city, country, emp_id(foreign key references Employee.emp_id)</pre> <pre>Employee: emp_id, name, addr_id(foreign key references Address.addr_id)</pre> <p>You must set both foreign keys in the EIS database as <b>on delete set null</b> or <b>on delete cascade</b>, regardless of the type of relationship you define for Address and Employee, or neither of them can be deleted.</p> </li> </ul>
<p>If the foreign key in a relationship points to a char(n) column, and it is the only column in the table, use the <b>rtrim(column name)</b> function in the MBOs column definition</p>	<p>This example has a relationship defined as:</p> <pre>States(Parent) --&gt; Sales_region(Child)</pre> <p>Sales_region contains one column named Region with a data type of char(7). The MBO definition should be:</p> <pre>SELECT rtrim(region) as region FROM sampledb.dba.sales_regions</pre>

## Validation Rules

Guideline	Example
MBOs used in a relationship must belong to the same Synchronization group	MBOs in a relationship need to share the same synchronization characteristics for proper synchronization of all MBOs within the relationship, therefore verify that all MBOs are in the same synchronization group.
By default, all MBOs in a relationship are assigned to the same cache group	You can assign different cache groups to MBOs within a relationship, however, doing so generates a warning message. You may want to, for example, separate cache groups with different cache policies for a <code>sales</code> MBO, which changes frequently, that has a relationship with the <code>catalog</code> MBO, which changes infrequently.
Unwired WorkSpace does not support Circular relationships	<p>Do not create circular relationships that work from parent-to-child then, eventually, back to the parent. For example:</p> <p>A relationship that includes three MBOs: A, B, and C, where</p> <pre>A-----&gt;B-----&gt;C-----&gt;A</pre> <p>is not supported.</p>



Guideline	Example
Composite relationship (composite graph) behavior	<ul style="list-style-type: none"> <li>• A child object refers to its parent via a foreign key reference from the child to a primary key value in the parent. When modeling your relationships, choose load parameters carefully to load children based on which parents are in the cache, so that foreign key references can always be resolved. At runtime Unwired Server must be able to resolve foreign key references. A poorly modeled relationship could have MBO instances pointed to by foreign keys that have not been loaded into the cache. Examples include: <ul style="list-style-type: none"> <li>• A composite parent that hasn't been loaded which results in the child not being inserted into the cache because composite orphans are not allowed – a child MBO can have multiple parents (many to one), but only one parent can have a composite relationship to the child: <p>In a composite relationship, when one child MBO does not have a synchronization parameter, and has multiple parents, each of which has a synchronization parameter, the child inherits synchronization behavior from both of the parents. If a composite relationship does not exist, Unwired WorkSpace randomly selects a parent through which the child synchronizes. Sybase recommends that you define one parent MBO as the composite parent to avoid unintended behavior. For example:</p> <p>There are two one-to-many relationships: A&lt;---&gt;&gt;C and B&lt;---&gt;&gt;C. Both MBOs A and B have a synchronization parameter while MBO C has none. In this case, one of the relationships should be a composite.</p> </li> <li>• An MBO modeling error where the relationship reference cannot be resolved results in an error message. For example: <ul style="list-style-type: none"> <li>• The Unwired Server log indicates a Null foreign surrogate key value where a child in a non-composite relationship points to a non-existent parent MBO, which could indicate that the MBO is a child in two separate relationships (one composite and one non-composite) and load parameters are improperly configured: <pre> Serious performance and data integrity issues may ensue. Consider modifying/re-sequencing cache definitions and/or load queries to ensure relationship integrity in the cache. </pre> </li> <li>• Sybase Control Center error log: <pre> Composite parent not found </pre> </li> </ul> </li> </ul> </li> <li>• If a parent MBO's primary key is auto-incremental and you want to perform multi-level insert operations, then the relationship must be composite. Otherwise the child MBO can only be created on an existing parent MBO.</li> <li>• When Composite is selected, all child operations are performed through the parent.</li> </ul>

Guideline	Example
	<ul style="list-style-type: none"> <li>• MBOs in a composite parent/child relationship where the child is loaded into the Unwired Server cache without a parent (composite orphans) are not supported. For example, a composite relationship between department(parent) and employees(child) that attempts to load employees for which there is no corresponding department is not allowed at runtime. In this scenario you could change the model to use a non-composite relationship or change the MBO model so the application always loads the parent entities before the children.</li> <li>• Foreign key attributes are updated automatically when setting a related parent/child object. Do not directly update the child's attribute (row) identified by the foreign key.</li> <li>• Multi-level insert (MLI) operations require composite relationships.</li> <li>• A composite graph should not span multiple load groups if loading from a structure-oriented EIS.</li> <li>• A composite graph should not span more than one endpoint definition.</li> </ul>
First/subsequent MBO relationship behavior	<p>Multiple MBOs created from a single operation result:</p> <ul style="list-style-type: none"> <li>• Subsequent MBOs as child entities (if the relationship mapping is based on a subsequent MBO load argument) are supported, and the relationship is treated the same as a relationship on an attribute, meaning it does not affect MBO loading.</li> <li>• Relationships from subsequent to first MBOs where the relationship mapping is based on the load argument of the first MBO are not supported.</li> </ul>
Local business objects	<ul style="list-style-type: none"> <li>• A relationship between two local business objects that does not have a primary key set is invalid.</li> </ul> <p>You cannot create relationships between local business objects using a composite primary key.</p>
Multi-level insert operation behavior	<p>Multi-level insert operations do not support unidirectional one to many relationships – newly created rows in the child MBO do not display prior to synchronization in a unidirectional one to many relationship. To view details prior to synchronization from the device application, go to the pending operations screen.</p>

Guideline	Example
Automatic setting of primary keys	Relationships require all primary keys on the source or target MBO (source or target depends on the relationship type) to be referenced in the relationship. If you design a relationship that does not include all the primary keys, a warning prompt allows you to decide whether to automatically set reference attributes to primary keys, and unset the unreferenced attributes primary keys. If you select <b>Yes</b> , Unwired WorkSpace automatically sets the primary key, if not, Unwired WorkSpace maintains the existing primary key setting, and displays a validation message according to the relationship rules.
Personalization key and default value removed from child parameter.	When creating a relationship from an attribute of a parent MBO to a load argument of a child MBO, the personalization key and default value cell of the load argument is disabled, and the personalization key and the default values, if any, are cleared, since those values are not used to load the child's value from the EIS at runtime.
It is a modeling error to have multiple relationships with the same set of links (every link has the same source and target attribute), for every relationship type (one-to-one, one to many, many to one, bidirectional, or composite), and generates a validation error.	For example, a relationship exists between customer and salesorder MBOs as: customer->id to salesorder->cust_id. The MBO developer cannot create another relationship using the same link: customer->id to salesorder->cust_id. Instead, either change the link or add new links to create the second relationship.

## Cache Validation Rules and Error Messages

These validation rules apply to cache groups and cache policies and are enforced during MBO development and, in some cases, when you deploy the MBO to Unwired Server. Messages are generated when validation rules are violated.

**Table 12. Cache group and cache policy validation rules and error messages**

Severity and message	Cause	Action
Error: Invalid interval for cache '<CacheGrp Name>': the value of interval must be between 1 and 2147483647 seconds.	The cache group's cache interval is out of range.	Specify a valid cache interval.
Error: Cache group name cannot be empty.	The cache group name is empty.	Enter a valid name.

## Validation Rules

Severity and message	Cause	Action
Error: Invalid cache name: '<CacheGrp Name>'. A valid identifier name must begin with an alphabetic character or an underscore, followed by alphanumeric characters or underscores.	An invalid cache name is specified.	Enter a valid name.
Error: Invalid cache name: '<CacheGrp Name>', name cannot be a Java or C# language.	The cache group name is a Java or C# keyword, such as binary, boolean, byte, char, date, datetime, time, decimal, double, float, object, int, integer, long, short, string, switch, and so on.	Do not use Java or C# keywords as cache group names.
Error: Invalid cache name: '<CacheGrp Name>', the length must be less than 64.	The length of the cache group name is longer than 64 characters.	Enter a valid name that is shorter than 64 characters.
Error: Duplicate cache group name: '<CacheGrp Name>'.	Another cache group uses the same name.	Enter a unique and valid name.
Warning: Relationship from 'MBO1' to 'MBO2' references to different cache groups. Mobile business objects with relationship should be grouped in the same cache group.	MBOs in a relationship are in different cache groups. This is allowed, but be aware that MBOs in different cache groups may use different cache refresh policies.	Move the MBOs within a relationship to the same cache group if they need to share the same data refresh schedule.
Warning: 'CREATE' operation '<Oper Name>' has 'Apply results to the cache' option set without returning primary key fields.	A create operation uses the "Apply results to the cache" cache policy, but the MBO to which it belongs does not have a primary key.	Set a primary key for the MBO in the Properties view from the <b>Attributes &gt; Attributes Mapping</b> tab.

Severity and message	Cause	Action
Warning: The output record of the 'CREATE' operation '<Oper Name>' with 'Apply results to the cache' option set does not contain the primary key columns.	The output of the CREATE operation with "Apply results to the cache" cache policy set does not cover the primary key columns.	Change the definition of the create operation to let its output to cover the primary key columns of the mobile business object. Note that this rule only works for Non-JDBC data source; for JDBC data source, as tooling can't identify the output metadata of an operation with executing it, user need to make sure he/she define the correct operation.
Error: It is unsupported to set runtime credential for mobile business object that belongs to a cache group using scheduled policy.	Set runtime credential for mobile business object which belongs to a cache group using scheduled policy.	Either move the mobile business object to cache group using on-demand policy or remove runtime credential.

## Synchronization Validation Rules and Error Messages

These validation rules apply to synchronization parameters and are enforced during MBO development, and, in some cases, when you deploy the MBO to Unwired Server or synchronize a device application. Messages are generated when validation rules are violated.

**Table 13. Synchronization validation rules and error messages**

Severity and message	Cause	Action
Error: Data type of the bound personalization key cannot map to the parameter: '<Para Name>'.	The datatype of the mapped personalization key differs from the datatype of the synchronization parameter, which may result in a synchronization failure.	Change the datatype of the personalization key to match the datatype of the synchronization parameter to which it maps.
Warning: Parameter data length is shorter than the bound personalization key's data length, value truncation might occur: '<Para Name>'.	Datatype length mismatch.	Change the datatype length of the parameter or that of the personalization key so they match.
Warning: Non-nullable parameter "ParaName" cannot map to the nullable personalization key: "PKName".	A nullable personalization is mapped to a non-nullable parameter.	Change the null-ability of the synchronization parameter or that of the personalization key so they match.

## Validation Rules

Severity and message	Cause	Action
Error: Invalid default value '<Default value>' for the '{1}' data type: '<Para Name>'.	The datatype of the default value does not match the given datatype.	Specify a valid default value.
Warning: Default value exceeds maximum length: '<Para Name>'.	The default value is too long compared to the datatype's length definition.	Enter a default value that is less than the maximum allowed.
Error: Binary parameter default value cannot exceed 16384 byte: '<Para Name>'.	The default value length is more than 16384 bytes.	Shorten the default value.
Error: The data types of the parameter '<Para Name>' and the 'Mapped to' attribute '<Attr Name>' should match.	Datatype mismatch.	Change the datatype of the synchronization parameter, or map it to a different attribute, or change the datatype of the attribute to which it maps.
Warning: The null-ability of the parameter '<Para Name>' should be the same as the attribute it is mapped to.	The null-ability of the parameter and the attribute to which it maps do not match.	Change the null-ability of the synchronization parameter or that of the attribute so they match.
Warning: The data length of the parameter '<Para Name>' is different from the data length of the its 'Mapped to' attribute '<Attr Name>'.	Datatype length mismatch.	Change the datatype length of the synchronization parameter or that of the attribute so they match.
Warning: No 'Mapped to' attribute specified for synchronization parameter: '<Para Name>'.	There is no <b>Map to</b> attribute defined for the parameter, which may lead to a synchronization failure.	Specify a <b>Map to</b> attribute for the synchronization parameter.
Error: Synchronization parameter '<Para Name>' of type {0}, does not allow length greater than {2}.	The synchronization parameter length is too long:  {0} may be STRING or BINARY  {2} 8191 for STRING 32767 for BINARY	Shorten the datatype length.
Error: Invalid parameter name: '<Para Name>'. A valid identifier name must begin with an alphabetic character or an underscore, followed by alphanumeric characters or underscores.	An invalid parameter name is specified.	Enter a valid name.

Severity and message	Cause	Action
Error: Invalid parameter name: '<Para Name>', name cannot be a Java or C# language keyword.	The parameter name is a Java or C# keyword, such as binary, boolean, byte, char, date, datetime, time, decimal, double, float, object, int, integer, long, short, string, switch, and so on.	Do not use Java or C# keywords as parameter names.
Error: Invalid parameter name: '<Para Name>', the length must be less than 100.	The length of the parameter name is longer than 100 characters.	Enter a name shorter than 100 characters.
Error: Duplicate parameter name: '<Para Name>'.	The name of the parameter is the same as an existing synchronization parameter.	Enter a valid name.
Error: Invalid '<Para Name>' name: name cannot be empty.	No parameter name has been specified.	Specify a valid name for the parameter.
Warning: The length of synchronization parameter '<Parameter name>' is not specified, set to <Default length> by default. Update the length to avoid truncation if necessary.	If the parameter's length is not specified, a default length is set.	Specify the length.
Error: Structure "<Structure name>" contains field "<attribute name>" with data type "<BigString/BigBinary>", and cannot be used as a client parameter "<client parameter name>".	A structured client parameter can not contain a field of data type BigString or BigBinary.	Change the parameter to use a structure that does not contain a large object data type.
Warning: The length of argument "<Argument name>" is unknown, so the length of client parameter "Client parameter" is set to <length> by default. Update the client parameter length to avoid data truncation if necessary.	The client parameter was given a default length when the argument length is unknown.	Update the client parameter length to avoid possible data truncation.

## **Object Query Validation Rules and Error Messages**

Validation rules are enforced during object query development, and, in some cases, when you deploy the MBO to Unwired Server. Messages are generated when validation rules are violated.

**Table 14. Object query validation rules and error messages**

<b>Severity and message</b>	<b>Cause</b>	<b>Action</b>
Error: Object query cannot have an empty name.	An empty string is specified as the object query name.	Enter a valid name.
Error: Invalid object query name: '<Query Name>'. A valid identifier name must begin with an alphabetic character or an underscore, followed by alphanumeric characters or underscores.	An invalid object query name exists.	Enter a valid name.
Error: Invalid object query name: '<Query Name>', name cannot be a Java or C# language keyword.	The object query name is a Java or C# keyword, such as binary, boolean, byte, char, date, datetime, time, decimal, double, float, object, int, integer, long, short, string, switch, and so on.	Do not use Java or C# keywords as object query names.
Error: Invalid object query name: '<Query Name>', the length must be less than 64.	The length of the object query name is longer than 64 characters.	Shorten the object query name.
Error: Duplicate object query name: '<Query Name>'.	The object query name is a duplicate.	Enter a valid and unique name.
Error: Duplicate name: '<Query Name>' in the mobile business object.	The object query name is the same as the MBO to which it belongs.	Enter a valid and unique name.
Error: The name: '<Query Name>' duplicates with an operation name.	The object query name is the same as an operation in an MBO to which they both belong.	Enter a valid and unique name.



Severity and message	Cause	Action
Error: The name: '<Query Name>' duplicates with an attribute name.	The object query name is the same as an attribute in an MBO to which they both belong.	Enter a valid and unique name.
Error: Query name '<Query Name>' is a reserved word.	The object query name is a reserved word: pull, downloadData, find, load, findPushList, _find, getPendingObjects, findWithQuery, getSize, getReplayPendingObjects, findSubscriptionsToBeNotified, and findBy.	Enter a valid name.
Error: Query definition cannot be empty.	The object query definition is empty.	Define a valid object query. <hr/> <b>Note:</b> Unwired WorkSpace validates only whether the object query is empty, but does not validate its syntax. <hr/>
Warning: No attribute is mapped to parameter '<Para Name>'.	The object query parameter has no attribute mapping.	Specify an attribute mapping for the parameter.
Error: The name: '<Query Name>' is a duplicate of the query name "FindAll".	User checks the box to generate 'FindAll' query and also defines a query named 'FindAll'.	Use another valid name or uncheck the box to generate 'FindAll' query.
Warning: Value truncation might occur, as the length of the parameter '<Para Name>' is shorter than the 'Mapped to' attribute's length in object query '<Query Name>'.	Data lengths do not match.	Change the parameter data length or the mapped attribute's data length so they match.
Error: Object query parameter "<Parameter name>" should not be mapped to attribute "<Attribute name>" with data type "<BigString/BigBinary>".	Object query parameter must not be mapped to a BigString/BigBinary attribute.	Change the attribute map to a non-BigString/BigBinary attribute or do not map the attribute at all.

Severity and message	Cause	Action
Warning: The length of query parameter <Query parameter name> is not specified, set to <Default length> by default. Update the length to avoid truncation if necessary.	The query parameter was set to a default length because the query length was not specified.	Update the query parameter length to avoid possible data truncation.

## Personalization Key Validation Rules and Error Messages

Validation rules are enforced when defining personalization keys, assigning them to attributes or parameters, and, in some cases, when you deploy the MBO to which the personalization key is assigned to Unwired Server. Messages are generated when validation rules are violated.

**Table 15. Personalization key validation rules and error messages**

Severity and message	Cause	Action
Error: Invalid personalization key name: '<PK Name>'. A valid identifier name must begin with an alphabetic character or an underscore, followed by alphanumeric characters or underscores.	The personalization key name is invalid.	Enter a valid name.
Error: Invalid personalization key name: '<PK Name>', name cannot be a Java or C# language keyword.	The personalization key name is the same as a Java or C# keyword, such as binary, boolean, byte, char, date, date-time, time, decimal, double, float, object, int, integer, long, short, string, switch, and so on.	Do not use Java or C# keywords as personalization key names.
Error: Invalid personalization key name: '<PK Name>', name cannot be a reserved personalization parameter name.	The name of the personalization parameter is a reserved key, such as: object, binary, boolean, byte, char, date, date-time, decimal, double, float, int, integer, long, short, string, time, cle.	Specify another name for the personalization parameter.
Error: Invalid personalization key name: '<PK Name>', the length must be less than 64.	The length of personalization key name exceeds the allowed maximum.	Enter a valid name that is less than 64 characters.

Severity and message	Cause	Action
Error: Duplicate personalization key name: '<PK Name>'. The name is case-insensitive.	A personalization key with the same name already exists.	Enter a valid and unique name.
Error: The default value(s) of a non-nullable personalization key should not be set to '<NULL>': '<PK Name>'.	A null default value has been set for a non-nullable personalization key.	Set the default value for the personalization key to a non-null value.
Warning: Default value for the '<Type>' data type for personalization key '<PK Name>' contains warning : '<Msg>'.	A structure personalization key contains a default value that generates a warning message.	Correct the problem identified by the warning message.
Error: Invalid default value '<Default Value>' for the '<Type>' data type of personalization key: '<PK Name>'.	The default value for the datatype of the personalization key is invalid.	Set the default value to conform to the datatype.
Error: Invalid < PK Name > name: name cannot be empty.	The <PK Name> name cannot be empty.	Enter a valid value for the name of the <PK Name>.
Error: Invalid structure default value for the '<Type>' data type for personalization key '<PK Name>': '<Msg>'.	A structure personalization key contains an invalid default value.	Correct the default value of the structure personalization key identified by the error message.
Warning: Default value exceeds allowed maximum length: '<PK_length>'.	The length of the default value exceeds the data type's maximum length.	Set the default value to conform to the data type length.
Error: Data type is not specified for the personalization key : "<PK_name>".	The data type is not specified for the personalization key.	Specify a proper data type for the personalization key.
Error: Structure type is missing for the personalization key : "<PK_name>".	The structure type for the personalization key is missing in the project.	Specify a proper structure data type for the personalization key.
Error: Structure "<Structure name>" contains field "<Attribute name>" with data type "<BigString/BigBinary>", and cannot be used as a personalization key.	A structure data type containing data type BigString or BigBinary has been defined as personalization key data type.	Change the structure data type to a non-BigObject-attribute structure or use any other simple type.

Severity and message	Cause	Action
Error: The length of personalization key "<PK name>" is not specified, set to <default length> by default. Update the length to avoid truncation if necessary.	The personalization key was set to a default length because the personalization key length was not specified.	Update the personalization key length to avoid possible data truncation.

## **Custom Filters, Roles, and Synchronization Group Validation Rules and Error Messages**

Validation rules are enforced during MBO development and, in some cases, when you deploy the MBO to Unwired Server. Messages are generated when validation rules are violated.

**Table 16. Custom filter, role, and synchronization group validation rules and error messages**

Severity and message	Cause	Action
Error: '<Filter class path>' has compile errors	A compilation error occurred in the result-set filter class file.	Correct the error in the filter class.
Warning: Duplicated result set filters in '<MBO Name>'. The filter class path '<Filter class path>' is shared by more than one filter.	One result-set filter is added to the same MBO more than once.	Remove duplicate filters.
Error: '<Filter class path>' does not implement <Interface name> interface.	The filter class does not implement the required interface.	Change the filter class to implement the required interface.
Error: '<Filter class path>' does not exist in workspace.	The filter class has been deleted outside of the workspace, and the related result filter class information has not been updated to reference the MBO.	Delete the result set filter from the MBO, or add the original filter class back to the workspace.
Error: Duplicate case insensitive synchronization group name: '<SyncGroup Name>'.	The synchronization group name is a duplicate of another synchronization group.	Enter a valid and unique synchronization group name.

Severity and message	Cause	Action
Error: The change detection interval must be between 1 and 2147483647 seconds for the synchronization group: '<SyncGroup Name>'.  	The synchronization group interval has an invalid value.	Enter a valid interval value.
Warning: Relationship from 'MBO1' to 'MBO2' references two Mobile business Objects in different synchronization Groups. Mobile business objects with relationship should be grouped in the same synchronization group.	MBOs in a relationship are grouped in different synchronization groups.	Resolve MBO dependencies and group MBOs with relationships in the same synchronization group.
Error: Synchronization group name '<SyncGroup Name>' is a reserved name.	The synchronization group uses the reserved name system or initialSync.	Rename the synchronization group.
Error: Invalid < object type > name: name cannot be empty.	The <object type> name cannot be empty.	Enter a valid value for the name of the <object type>.
Warning: Mobile business objects in the same synchronization group should have the same role assignments: '<SyncGroup Name>'.  	Mobile business objects in the same synchronization group have different role assignments.	Make sure the MBOs in one synchronization group have the same role assignments.
Error: Duplicate role name: '<Role Name>'.  	Duplication of role names.	Enter a valid and unique role name.

## Validation Rules

Severity and message	Cause	Action
Error: Invalid role name: '<Role Name>', name cannot be a Java or C# language keyword.	The role name is the same as a Java or C# keyword, such as binary, boolean, byte, char, date, datetime, time, decimal, double, float, object, int, integer, long, short, string, switch, and so on.	Do not use Java or C# keywords as role name.
Error: Invalid personalization key name: '<Role Name>'. A valid identifier name must begin with an alphabetic character or an underscore, followed by alphanumeric characters or underscores.	An invalid role name exists.	Enter a valid role name.

## Mobile Workflow Validation Rules and Error Messages

Validation rules are enforced when defining mobile workflows. Messages are generated when validation rules are violated.

**Table 17. Mobile workflow validation rules and error messages**

Severity and message	Cause	Action
problem.empty.module.name=Module name cannot be empty	The Flow Design page property Module name requires an entry, but the property is blank.	Provide a module name.
problem.module.name.too.long=The length of the module name cannot exceed 100 characters	The Module name is too long.	Shorten the module name.
problem.module.version.too.low=The module version must be at least 1	The Flow Design page property Module version requires an entry, but the property is blank.	Enter a module version of at least 1.

Severity and message	Cause	Action
problem.match.rule.invalid.regex=The regular expression "{0}" for the "{1}" match rule is not valid.	The regular expression for the match rule indicated in the message is not valid.	Supply a regular expression that is a valid.
problem.screen.key.duplicated=The screen key "{0}" has been duplicated.	The screen key indicated in the messages is already used. The screen key value must be unique.	Enter another screen key that is unique.
problem.empty.screen.name=Screen name cannot be empty.	The Screen name property requires an entry, but the property is blank.	Enter a screen name.
problem.empty.screen.key=Screen key cannot be empty.	The Screen key property requires an entry, but the property is blank.	Enter a screen key.
problem.control.has.no.key=The control "{0}" on screen "{1}" has no key.	The control for the screen identified in the message requires a key, but the property is blank.	Enter a key for the control.
problem.control.uses.key.with.unsupported.datatype=The control "{0}" on screen "{1}" uses a key with a data type that this type of control does not support.	The data type entered for the control on the screen indicated in the message is not supported.	Enter a data type that is valid for the control.
problem.control.uses.key.used.by.another.control=The control "{0}" on screen "{1}" uses a key that another control on the same screen is using.	The key for the control and screen indicated in the message is already used by another control. The key must be unique.	Enter another control key that is unique.
problem.control.is.choice.but.has.no.options=The choice control "{0}" on screen "{1}" has no options.	The choice control requires one or more selection items. No items are included.	If you intend to use the choice control, select items to add from the drop-down list. Note that you can supply the selection items either statically or dynamically, but you must supply them. Or, select another control type.

## Validation Rules

Severity and message	Cause	Action
problem.control.is.attachmentviewer.but.has.no.mimetype.or.filename=The attachment viewer control "{0}" on screen "{1}" has neither its MIME type nor its file name properties configured.	The attachment viewer control and screen named in the message does not include an attachment.	Configure either a MIME type or a file name to identify the attachment.
problem.too.many.default.menu.items=The screen "{0}" has more than one default menu items.	The screen identified in the message has multiple default menu items, but only one is allowed.	Identify only one menu item as the default.
problem.open.menu.item.with.no.screen=The menuitem "{0}" on the screen "{1}" is marked as being of type Open but does not have its screen property set.	The menu item on the screen identified in the message does not have screen properties set to correspond with the Open type.	Provide the screen to which to link.
problem.empty.flow.starting.point.name=Flow starting points cannot have an empty name.	The flow design requires starting point coordinates, but none are assigned.	Provide the starting point properties.
problem.operation.parameter.unmapped=No key is mapped to parameter {0}->{1}()->{2}	A key is required for the operation parameter indicated in the message.	Provide a key for the operation parameter.
problem.namedquery.parameter.unmapped=No key is mapped to parameter {0}->{1}()->{2}	A key is required for the named query parameter indicated in the message.	Provide a key for the named query parameter.
problem.key.unmapped=The key "{0}" is unused.	The key identified in the message is not used.	Either delete the key or map it to the required parameter.
problem.screen.has.no.flow.starting.point=The screen "{0}" is not linked to any starting points and thus will be unused.	The screen identified in the message is not linked to any starting parts so is not used.	Either delete the screen or map it to the required starting points.



Severity and message	Cause	Action
problem.action.has.pk.mapping.that.is.bound.to.nonexistent.pk=The screen "{0}" has a menu item "{1}" that has a personalization key mapping that refers to a personalization key, "{2}", that does not exist.	The personalization key that is identified in the messages is not mapped correctly.	Map to an existing personalization key.
problem.key.has.no.binding=The key "{0}" in the starting point "{1}" is not bound to any data.	The key identified in the message is not bound to a data source.	Map the starting point key to the data source.
problem.key.is.bound.to.nonexistent.pk=The key "{0}" in the starting point "{1}" is bound to a personalization key, "{2}", that does not exist.	The starting point key named in the message is bound to a personalization key that does not exist.	Either bind the key to an existing personalization key, or create the missing personalization key.
problem.key.in.non.server.initiated.starting.point.is.sent.from.server=The key "{0}" in the starting point "{1}", which is not server-initiated, is marked as being sent from the server, which is not supported.	The starting point key is not server-initiated, but it is associated with a server. Server-initiated starting point is not supported.	Change the starting point key to be non-server initiated.
problem.add.action.must.specify.a.screen=The menu item named "{0}" of type add on the screen "{1}" must specify a screen	The menu item on the screen identified in the message requires a screen, but one is not specified.	Specify a screen.
problem.key.name.duplicated=The starting point "{0}" has more than one key with the name "{1}"	The starting point named in the message has more than one key identified with the same name. A unique name is required for each key.	Rename any duplicate keys so all key names are unique.
problem.mismatched.match.rule.cc=The starting point "{0}"'s CC property does not match the workflow application's CC match rule.	The starting point named in the message for the e-mail header CC property does not match the CC match rule.	Check the starting point CC rule against the workflow application CC match rule, and make corrections so they match.

## Validation Rules

Severity and message	Cause	Action
problem.mismatch-ed.match.rule.from=The starting point "{0}"s From property does not match the workflow application's From match rule.	The starting point named in the message for the From property does not match the From match rule.	Check the starting point From rule against the workflow application From match rule, and make corrections so they match.
problem.mismatch-ed.match.rule.subject=The starting point "{0}"s Subject property does not match the workflow application's Subject match rule.	The starting point named in the message for the Subject property does not match the Subject match rule.	Check the starting point Subject rule against the workflow application Subject match rule, and make corrections so they match.
problem.mismatch-ed.match.rule.to=The starting point "{0}"s To property does not match the workflow application's To match rule.	The starting point named in the message for the To property does not match the To match rule.	Check the starting point To rule against the workflow application To match rule, and make corrections so they match.
problem.credential.request.missing.username.key=The Credential Request starting point does not have a Credential Cache Username key	The starting point for Credential Request does not have a Credential Cache Username key assigned, and one is required.	If credentials are required, assign a Credential Cache Username key.
problem.credential.request.too.many.username.keys=The Credential Request starting point has too many Credential Cache Username keys	The starting point for Credential Request has more than one Credential Cache Username keys assigned, and only one is required.	Identify one Credential Cache Username key, and remove all others.
problem.credential.request.missing.password.key=The Credential Request starting point does not have a Credential Cache Password key	The starting point for Credential Request requires a Credential Cache Password key.	Specify a Credential Cache Password key.
problem.credential.request.too.many.password.keys=The Credential Request starting point has too many Credential Cache Password keys	The starting point for Credential Request has more than one Credential Cache Password keys assigned, and only one is required.	Identify one Credential Cache Password key, and remove all others.

Severity and message	Cause	Action
<p>problem.missing.credential.cache.username.or.password=Please specify which editboxes are the credential cache username and password.</p>	<p>The EditBoxes for cache username and password are not clearly identified.</p>	<p>Identify the cache username and password EditBoxes clearly by setting the appropriate property in the property sheet of the username and password edit boxes.</p>
<p>problem.credential.request.screen.invokes.mbo=The screen pointed to by the Credential Request starting point should be used only to set the credential request user name and password.</p>	<p>The Credential Request starting point is set to an MBO. It should be set to a login screen.</p>	<p>Identify a login screen for entering log in credentials.</p>
<p>problem.credentials.are.not.cached.for.email.triggered.workflows=The credentials used in an notification triggered workflow are not cached between invocations. Consider using static credentials for notification triggered workflows.</p>	<p>You have specified dynamic credentials for the workflow, but static credentials may be more appropriate.</p>	<p>Consider using static credentials, where all users use the same login information, for notification triggered workflows.</p>
<p>problem.missing.critical.screen.flow=At least one Server-initiated or Client-initiated flow starting point needs to have a screen flow.</p>	<p>The screen flow design must include at least one server- or client-initiated flow starting point. One is not defined.</p>	<p>Include a server- or client-initiated flow starting point.</p>
<p>problem.operation.noinstance=The screen "{0}" has a menu item "{1}" which has one or more parameters that are either unmapped or are mapped to attributes of the "{2}" MBO without having an instance of that MBO.</p>	<p>The screen and menu item named in the message is not complete. Parameters are either unmapped or incorrectly mapped to an MBO.</p>	<p>Complete the menu mapping task.</p>
<p>problem.namedquery.noinstance=The screen "{0}" has a menu item "{1}" which is trying to invoke the object query "{2}" on the "{3}" MBO without having an instance of that MBO.</p>	<p>The screen and menu item named in the message cannot invoke an object query because the MBO instance is not included.</p>	<p>Determine why the MBO instance is not included, then fix the problem.</p>

## Validation Rules

Severity and message	Cause	Action
problem.listview.operation.no.params.bound.to.attributes=The screen "{0}" has a menu item "{1}" which is invoking an operation "{2}" and hasn't bound any of its parameters to attribute keys, which is not recommended.	The screen and menu item named cannot invoke a listview operation. Parameters are not bound to attribute keys.	Bind parameters to attribute keys, then retry the operation.
problem.no.credential.request.screen=This application, as designed, needs a Credential Request screen and does not have one.	The workflow application requires a Credential Request screen.	Add a Credential Request screen.
problem.no.credential.request.starting.point=This application, as designed, needs a Credential Request starting point	The workflow application requires a Credential Request starting point.	Add a Credential Request starting point.
problem.key.bound.to.relationship.that.is.many.to.one=The key "{0}" is bound to a relationship that is marked as being many-to-one. Many-to-one relationships are not supported.	The key is bound to a many-to-one relationship, which is not supported for mobile workflow applications. Only composite relationships are supported.	Change the key to bind to a composite relationship.
problem.key.bound.to.non.composite.relationship=The key "{0}" is bound to a relationship that is not marked as being composite. Only composite relationships are supported.	The key is bound to a relationship other than composite. Only composite relationships are supported.	Change the key to bind to a composite relationship.
problem.key.bound.invalid.relationship=The key "{0}" is bound to a relationship, "{1}", on the MBO "{2}", but that relationship no longer exists.	The key is bound to an MBO relationship that no longer exists.	Identify an MBO relationship that exists.
problem.first.screen.in.flow.has.save=The screen "{0}", the first screen in the "{1}" flow, has a menuitem, "{2}", which is of type Save. Save menuitems are not supported on the first screen of the flow.	The screen named and the first screen in the flow identified have the same Save menuitem. Save menuitems are not supported on the first screen of the flow.	Change the design so that Save menuitems are not used on the first screen of the flow. Consider using a Cancel, Close, or Submit workflow instead.

Severity and message	Cause	Action
problem.first.screen.in.flow.okaction.is.save=The screen "{0}", the first screen in the "{1}" flow, has its okaction property set to Save. Save actions are not supported on the first screen of the flow.	The named screen and first screen in the flow both have the okaction property set to Save. Save actions are not supported on the first screen of the flow.	Change the design so that the okaction property is set to Cancel on the first screen of the flow.
problem.mbo.not.found=The MBO "{0}" referenced from "{1}" could not be found	The MBO named in the reference could not be found.	Check the design to make sure the MBO reference is correct. You can use the MBO References dialog to retarget existing unfound MBO references to existing MBOs.
problem.mbo.not.found.on.email.starting.point=The MBO "{0}" referenced from the notification trigger could not be found	The MBO named in the reference could not be found.	Check the design to make sure the notification trigger reference is correct. You can use the MBO References dialog to retarget existing unfound MBO references to existing MBOs.
problem.key.mbo.attribute.not.found=The key "{0}" references a non-existent attribute, "{2}", of the MBO "{1}"	The key refers to an attribute that does not exist for the MBO named in the message.	Correct the key to refer to the correct attribute, or correct MBO with the attribute.
problem.fs.key.is.also.cc.key=The key of the editbox "{0}" from the screen "{1}" is a cached credential and flow-specific key.	Information only. The key identified with the editbox is a cached credential and flow-specific key.	None.
problem.default.value.is.not.numeric=The default value "{0}" for the control "{1}" on the screen "{2}" is an invalid numeric value.	The default value specified for the control on the screen is an invalid numeric value.	Change the default to a valid numeric value.
problem.key.type.does.not.match.option.type=The key type of "{0}" is not compatible with the control's logical type setting of "{1}" for the control "{2}" on the screen "{3}".	The key type specified is not valid for the control's logical type setting.	Specify a valid key type that is compatible with the control's logical type setting.

## Validation Rules

Severity and message	Cause	Action
problem.menuitem.action.cant.be.asynchronous=The screen "{0}" has a menu item "{1}" which is of type Submit Workflow under conditions in which only online requests are supported.	This menu item cannot be bound to the "Submit Workflow" action type under the current design condition.	Change the design to allow use of the "Submit Workflow" action type for this menuitem, or use the "Online Request" action type for the menuitem.
problem.menuitem.duplicate.name=The screen "{0}" has two or more menu items with the same name, "{1}".	The screen named has duplicate menu items.	Give both menu items unique names.
problem.add.menuitem.empty.screen=The screen "{0}" has a menu item "{1}" which is of type Open and doesn't have a valid screen property setting.	A menu item on the named screen does not have valid screen property settings for the Open type.	Provide the correct property setting.
problem.add.menuitem.empty.screen=The screen "{0}" has a menu item "{1}" which is of type Add and doesn't have a valid screen property setting.	A menu item on the named screen does not have valid screen property settings for the Add type.	Provide the correct property setting.
problem.choice.control.refers.to.key.that.isnt.accessible=The choice control "{0}" on screen "{1}" refers to a key, "{2}", which it doesn't have access to.	The choice control on the named screen refers to a key that it does not have access.	Either change the control to refer to an accessible key, or arrange to have access to the key.

# Glossary: Sybase Unwired Platform

Defines terms for all Sybase® Unwired Platform components.

**4GL** – Short for 4th generation language, this is a term generally applied to development methods and tools that provide graphical interfaces for software development, reducing or eliminating the need to work directly with source code to implement business logic.

**activation code** – a method of identifying the user in the runtime via code, rather than by authenticated identity. This is considered a manual connection registration method. During the connection registration a connection might be activated when using an activation code.

**administration perspective** – or administration console. The Unwired Platform administrative perspective is the Flash-based Web application for managing Unwired Server. *See* Unwired Platform.

**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 Sybase Control Center administration roles to avoid having to authenticate to Unwired Platform in addition to Unwired Server:

- a domain administrator requires only the "sccUserRole" role.
- a platform administrator requires both the "sccAdminRole" and "sccUserRole" roles.

**Adobe Flash Player** – a third party software tool required to run Unwired Platform. Because Adobe does not support 64-bit browsers you must run Unwired Platform in a 32-bit browser.

**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.

**Agentry application hierarchy** – The order and structure of, and the relationship between, the various definition types that comprise any Agentry application project.

**Agentry application project** – The definitions and references to synchronization logic that comprise the encapsulated business logic of an Agentry mobile application.

**Agentry Client** – The executable client software within the Agentry paradigm that consumes and processes the business logic defined in the Agentry application project, and presents that business logic and behavior to the mobile user.

**Agentry Client Branding SDK** – A toolset provided to allow partners and customers developing Agentry applications to brand those applications for packaging and delivery.

**Agentry Editor** – The primary development tool for an Agentry application project, built as a plug-in to the Eclipse IDE and presenting a 4GL development interface to define and modify business logic and rules within the application project.

**Agentry Server** – The component of the Agentry paradigm that interfaces between the Agentry Clients and the back end system or systems with which the mobile application synchronizes data. The Agentry Server is also responsible for serving up the business logic of the mobile application to the Agentry Clients during the initial implementation and in subsequent updates to the application logic.

**Agentry Test Environment (ATE)** – An Agentry Client wrapped in numerous testing, debugging and monitoring tools, used primarily by developers to mimic various client device platforms during the development life cycle of a mobile application.

**anonymous user** – a user type who can access the system without identification.

**Apple Push Notification Service (APNS)** – service provided by Apple for devices running the iOS operating system. The APNS acts as a intermediary to push notifications from the provider to the device rather than have the application operate as an active listener for those notifications.

**application** – in Unwired Server (and visible in Unwired Platform), the runtime entity that can be directly correlated to a native or Hybrid App. The application definition on the server establishes the relationship among packages used in the application, the domain to which the application is deployed, the user activation method for the application, and other application-specific settings. *See* client application and mobile application.

**application activation** – from the Sybase Unwired Platform standpoint, all activities that allow an identified or anonymous user to be paired with an application and its connections and customizations. Or, all activities that allow an application to be activated. An activated application creates an instance that is known by Unwired Platform.

From the application user standpoint, application activation is the automated series of events by which a user, without administrative intervention, can start consuming services.

**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 Data** – The data that is the business logic served up by the Agentry Server to the Agentry Client during a transmit; term used to distinguish between this data and Production Data.

**application ID** – the unique ID that identifies an application (automatic or manual)

**application instance** – represents a client application on a single device.



**application node** – in Unwired Platform, a registered application with a unique ID. The main entity that defines the behavior of device and back end interactions.

**application provisioning** – placing a client application on a device which includes:

1. Copying the application to the device.
2. Installing the application on the device.
3. Configuring the application on the device.
4. Securing the device.

**application registration** – configuring an application to work with Sybase Unwired Platform. Registration requires a unique identity that defines the properties for the device and back-end interaction with Unwired Server.

**application server node** – an Unwired Server configuration that can process all of the application types that are supported by Unwired Platform. An application server node includes the installation of Sybase Control Center to allow administration of the complete cluster. See *scale-out node*.

**application user** – the distinct set of identities (identified or anonymous) that have ever been in contact with the system by utilizing the application. In Sybase Control Center, an application user is the distinct list of names under which a user has been identified to the system. An application user may also be a user (identified or anonymous) that has been associated with an application ID.

**artifacts** – client-side or automatically generated files; for example: `.xml`, `.cs`, `.java`, `.cab` files.

**availability** – indicates that a resource is accessible and responsive.

**binary large object (BLOB)** – a collection of binary data stored as a single entity in a database management system. A BLOB may be text, images, audio, or video.

**Business Application Programming Interface (BAPI)** – 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.

(DOE) refers to a BAPI "wrapper" that contains the parameters required to synchronize the device data and the SAP back-end data.

**cache database (CDB)** – stores runtime metadata (for Unwired Platform components) and cache data (for MBOs). The CDB is also referred to as the Unwired Server cache, or just cache. See data tier.

**cache group** – defined in Unwired WorkSpace, a group of MBOs to which 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).

**Change and Transport System (CTS)** – SAP Change and Transport System for Advanced Business Application Programming (ABAP). CTS enables administrators to distribute artifacts and automate deployment to different target systems that are connected through transport routes. The enhanced version of this product (CTS+) can be used for transporting non-ABAP objects such as Sybase Unwired Platform development artifacts.

**client application** – in Unwired Platform, the software that runs on a smart phone, tablet computer, or other mobile device. *See* mobile application.

**client device** – General term used to describe the device upon which the Agency Client is installed and running; can apply to any of the devices to which the Agency Client can be installed, including PCs, laptops, tablets, and smart phones.

**client object API** – described in *Developer Guide: BlackBerry Object API Applications*, *Developer Guide: iOS Object API Applications*, and *Developer Guide: Windows and Windows Mobile Object API Applications*.

**cluster** – also known as a server farm. Typically, clusters are set up 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 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.

**command line interface (CLI)** – the standard term for a command line tool or utility.

**configuration cache** – distributed memory configuration used by all cluster members.

**connection** – configuration details and credentials required to connect to a database, Web service, or other EIS.

**connection pool** – 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. Or a collection of proxy connections pooled for their respective back ends, such as SAP Gateway

**connection profile** – in Unwired WorkSpace, the configuration details and credentials required to connect to an EIS.

**consolidated data store (CDS)** – the SAP Data Orchestration Engine (DOE) middleware component where data is consolidated by the DOE.

**context variable** – in the Unwired WorkSpace Hybrid App Designer, context variables are automatically created when a developer adds references 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 arguments to use the dynamic value of a selected record of an MBO during runtime.

**data change notification (DCN)** – allows an EIS to synchronize its data with the cache database through a push event.

**data manipulation language (DML)** – a group of computer languages used to retrieve, insert, delete, and update data in a database.

**Data Orchestration Engine (DOE)** – the SAP component that supports data exchange between SAP back-end systems and mobile devices.

**Data Orchestration Engine Connector (DOE-C)** – the Sybase component that connects applications running on Sybase Unwired Platform with the DOE.

**data points** – using performance harnesses and measuring KPIs, specific points defined to take and report measurements. Data points are required to implement end-to-end tracing features.

**data refresh** – synchronizes data between the cache database and a back-end EIS so that data in the cache is updated. *See* scheduled data refresh.

**datasource** – in Unwired WorkSpace, the persistent-storage location for the data that a mobile business object (MBO) can access.

**data synchronization** – the process of establishing consistency among data from a source to a target data storage and vice versa, and the continuous harmonization of the data over time.

**data tier** – includes Unwired Server data such as the cache database (CDB), cluster, monitoring, and messaging databases.

**data vault** – a secure store across the platform that is provided by an Unwired Platform client.

**Definition** – A finite component of an Agency application project, encapsulating a general functional type within an application, and that exists within the Agency application hierarchy and may have both a parent definition and one or more child definition types, as well as attributes that comprise its makeup.

**demilitarized zone (DMZ)** – 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.

**deploy** – 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, 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** – an XML file that contains role-mapping and domain-connection information, and describes how to deploy a deployment unit to Unwired Server. 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** – the mode in which a mobile application project or mobile deployment package is deployed to the target Unwired Server.

**deployment profile** – a named instance of predefined server connections and role mappings that allows developers to automate deployment of multiple packages from 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 output generated by the Unwired WorkSpace build process. It enables a mobile application to be installed and used in a preproduction or production environment. A deployment unit allows deployment of all required objects, logical roles, personalization keys, and server connection information together, without access to the development project. You can deliver a deployment unit and a deployment descriptor—jointly called a deployment archive—to an administrator for deployment.

**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** – a software application that runs on a mobile device. *See* mobile application.

**device notification** – received by replication synchronization clients 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 provisioning** – making an out-of-the box corporate device or bring your own device (BYOD) secure and ready for synchronization.

**device user** – the user identity tied to a device.

**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 Unwired Platform; only features and domains they can manage are visible.

**domains** – provide a logical partitioning of a hosting organization's environment, so that the organization achieves increased flexibility and granularity of control in multi tenant

environments. By default, the Unwired Platform installer creates a single domain named "default." However, the platform administrator can also add more domains as required.

**enterprise information system (EIS)** – a back-end system, such as a database.

**end-to-end tracing** – supportability feature that allows developers to add specified libraries and code to applications to enable tracing during runtime.

**Enterprise Explorer** – in Unwired WorkSpace, allows you to define data sources and view their metadata (schema objects in case of database, BAPIs for SAP, and so on).

**entity set definition for mobile applications (ESDMA)** – contains the external metadata definition of a software component version (SWCV) you can use to construct a client that can interact with the DOE.

**ESDMA converter** – the Sybase Unwired Platform utility that converts an ESDMA bundle into an Unwired Platform package that can be deployed to one or more instances of Unwired Platform.

**export** – the Unwired Platform administrator can export mobile objects, then import them to another server on the network.

**Gateway for NetWeaver Mobile** – add-on that is installed on top of DOE to provide integration with Sybase Unwired Platform.

**Google Cloud Messaging (GCM)** – a free service for sending messages to Android devices. GCM requires an API Key to allow Unwired Server to send push notifications over GCM.

**hostability** – *See* multitenancy.

**Initial Transmit** – The term applied to the first transmit performed by an Agency Client which, at the onset of the transmit, contains no business logic.

**integrated development environment (IDE)** – a set of programs that provide comprehensive facilities for software development. An IDE is typically associated with a specific programming language and provides features for authoring, modifying, compiling, deploying and debugging software from a single user interface.

**Introscope** – a third-party tool that can be integrated into a system landscape to quickly isolate and resolve performance issues wherever they arise in each stage of the application life cycle.

**Java development environment (JDE)** – IDE specific to the Java programming language that is used in Unwired Platform to create and test BlackBerry Java applications.

**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* truststore.

**Lightweight Directory Access Protocol (LDAP)** – an application protocol for accessing, querying, and modifying data in distributed directory services.

**load group** – a set of MBOs that load into the Unwired Server cache from the EIS from the same shared read operation.

**LoadRunner** – a performance testing tool developed by Hewlett Packard (HP). Sybase Unwired Platform provides an extension which integrates with LoadRunner to perform load testing for Hybrid Web Container and Open Data Protocol client applications.

**local business object** – defined in Unwired Workspace, objects that are used as local a data store on a device. Local business objects are not bound to EIS data sources, so cannot be synchronized.

**logical role** – defined in MBOs, and mapped to physical roles when the deployment unit that contains the MBOs are deployed to Unwired Server.

**matching rules** – a rule that triggers a Hybrid App. Matching rules are used by the Hybrid App e-mail listener to identify e-mail messages that match the rules specified by the administrator. When an e-mail message matches the rule, Unwired Server sends the message as a Hybrid App to the device that matches the rule. A matching rule is configured by the administrator in Unwired Platform.

**MBO attribute** – 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** – links MBO attributes and operations to a physical data source through a connection profile.

**MBO composite object graph** – (composite graph) refers to a group of entities (MBOs) in a composite relationship. Entities in a composite relationship are owned by their composite parents such that each entity in a graph has at most one composite parent. Entities that are not part of a composite relationship are treated by Unwired Server as a graph consisting of a single root node.

**MBO composite operation** – (composite operation) refers to a create, update, or delete operation defined on the root MBO of a composite graph that accepts the entire composite graph as input.

**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** – 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

**message-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* synchronization.

**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.

**Messaging framework** – the messaging functionality provided by the Sybase SAP DOE Connector to connect the SAP DOE with devices.

**mobile application** – (mobile app) is a software application designed to run on smart phones, tablet computers and other mobile devices. For Agentry, a general term used to refer to the mobile application built in Agentry and generally used to make the fine distinction between the Agentry Client executable itself, and the business logic which it is processing and presenting to the user; in essence this business logic is the mobile application.

**Mobile Application Diagram** – the graphical interface for creating and editing MBOs. By dragging and dropping a datasource 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.

**Hybrid App packages** – deployed to Unwired Server, and can be deployed to mobile devices, via the Unwired Platform administrative perspective in Unwired Platform. Hybrid App packages use the messaging synchronization model.

**mobile business object (MBO)** – the fundamental unit of data exchange in Sybase Unwired Platform. An MBO roughly corresponds to a data set from a back-end datasource. 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 that determine the manner in which the client passes information to the EIS.

You can define relationships between MBOs, and link attributes and parameters in one MBO to attributes and parameters in another MBO.

**Mobile data model** – shows the relationship between back-end enterprise data and the data on a mobile device.

**monitoring** – an Unwired Platform feature that allows administrators to identify areas of weakness or periods of high activity in a particular area. 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** – 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* 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.

**OData for SAP** – provides SAP Extensions to the OData protocol that enables users to build user interfaces for accessing the data published via OData. The interfaces require human-readable, language-dependent labels for all properties and free-text search within collections of similar entities and across (OpenSearch).

**OData metadata document** – describe the entity data model (EDM) for a given service, which is the underlying abstract data model used by OData services to formalize the description of the resources it exposes.

**OData Schema** – defines the structure of the XML files in the OData service.

**OData Service Document** – a document that describes the location and capabilities of one or more collections.

**Online Data Proxy (ODP)** – a light-weight edition of the Sybase Unwired Platform that provides a robust mobile infrastructure for enterprise IT organizations to securely roll out and manage the deployment of light-weight applications in a controlled and monitored approach.

**Open Data (OData) Protocol** – Web protocol for querying and updating data. It applies and builds upon Web technologies such as HTTP, Atom Publishing Protocol (AtomPub) and JSON to provide access to information from a variety of applications.

**operation** – *See* MBO operation.



**package** – 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 graphical interface view from which you can add MBOs, local business objects, structures, relationships, attributes, and operations to the Mobile Application Diagram.

**parameter/argument** – a value that is passed to an operation/method to determine the output. When you create an MBO, you can map MBO parameters/arguments to datasource arguments. For example, if a datasource 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:

- 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 parameter values that map to remote operation arguments and determine how the client passes information to the EIS.

**persistence layer** – the collection of database files, containing tables, that is available on the device through the Sybase Client Object API.

**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 set at the package level and also provide operation parameter values. There are three type of personalization keys: transient, client, and server.

Personalization keys 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 Unwired Platform 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. 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 - for example, by location, department, or project.

**physical role** – a security provider group or role that is controls access to Unwired Server resources.

**Problems view** – displays errors and warnings for the Mobile Application Project in Eclipse. This is a valuable source for collecting troubleshooting information and resolving issues during the MBO development phase, and avoiding device application problems later, for example, device application synchronization or data refresh errors.

**Production Data** – The data stored either in the back end system or on the client device that is synchronized between those two components during a transmit; term used to distinguish between this data and the Application Data.

**provisioning** – *See* application provisioning and device provisioning.

**Publish** – The term that describes the deployment and transformation of the business logic built and/or modified within the Agentry Editor to the Agentry Server with the intent of serving that business logic to Agentry Clients when they next synchronize.

**pull synchronization** – 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** – 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** – a list of pending activities, made up of in-flight messages for a messaging application. The server sends messages to specific destinations based on message order in the queue. The depth of the queue indicates how many messages are waiting to be delivered.

**recovery** – performing the activities required to bring a system to a usable/functional state after a failure (populating CDB, initializing client, and so on).

**relationship** – *See* MBO relationship.

**Relay Server** – *See* Sybase Hosted Relay Service.

**Relay Server Outbound Enabler (RSOE)** – an application that manages communication between Unwired Server and a relay server.

**remote function call (RFC)** – used 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.

**Representational State Transfer (REST) web services** – a style of software architecture for distributed hypermedia systems, such as the World Wide Web.

**resource** – a unique Sybase product component (such as a server) or a subcomponent.

**restoration** – returning a system to its prefailure state using one or more methods of recovery. Restoration does not guarantee a return to a usable state.

**role** – controls access to Sybase Unwired Platform resources. *See* logical role and physical role mapping

**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).

**runtime server** – an instance of Unwired Server that is running. Typically, a reference to the runtime server implies a connection to it.

**SAP** – one of the EIS types that Unwired Platform supports. SAP Business Suite applications (such as ERP, CRM, SRM, SCM, Industry Solutions and so on) consist of many technologies and components. Unless stated otherwise, the term “SAP” means a backend business application that is based on the SAP NetWeaver ABAP application server, for example ECC 6.0.

**Sybase Control Center** – in Unwired Platform, a Web-based interface that allows you to administer your installed Sybase products.

**SAP Hosted Relay Service** – a Web-hosted Relay Server that enables you to test your Unwired Platform development system.

**SAP License Audit** – a system measurement tool that analyzes usage data from Sybase Unwired Platform and SAP applications deployed to the Unwired Platform server.

**SAP Messaging Service** – the synchronization service that facilitates communication with device client applications.

**Sybase Unwired Platform** – a development and administrative platform that enables you to mobilize your enterprise. With Unwired Platform, you can develop MBOs in the Unwired WorkSpace development environment, connect to structured and unstructured data sources, develop mobile applications, and deploy mobile business objects and applications to Unwired Server, which manages messaging and data services between your data sources and your mobile devices.

**Unwired Platform X.X Service** – provides runtime services to manage, monitor, and control distributed Sybase resources. The service must be running for Unwired Platform to run. Previously called Sybase Unified Agent.

**Unwired WorkSpace** – Sybase Unwired Platform includes Unwired WorkSpace, which is a development tool for creating mobile business objects and mobile applications.

**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.

**SAP NetWeaver Gateway** – enables people-centric applications to consume SAP Business Suite data through popular devices and platforms in an easy and standards-based fashion.

**SAP Passport** – medium for transporting technical data in a request from the client to the server. Used for collecting trace and reporting information for chains of requests (RFC, HTTP) across system borders.

**SAP Solution Manager** – SAP standard platform for Application Life Cycle Management (ALM). SAP Solution Manager is a centralized, robust solution management toolset that facilitates technical support for distributed systems - with functionality that covers all key aspects of solution deployment, operation, and continuous improvement.

**scale-out node** – an Unwired Server configuration that is specifically designed to allow the stateless request/response HTTP and synchronous message services over an open HTTP channel to be scaled horizontally, by adding more server nodes. *See application server node.*

**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 Unwired Platform.

**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 an MBO, he or she can schedule a data refresh 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 an administrator to 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 Unwired Platform. A security configuration contains:

- A set of configured security providers (for example, LDAP) to which authentication, authorization, and attribution are delegated.
- Role mappings which can be specified at the domain or package level.

**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 for component communication.

**security provider** – a security provider and its repository holds information about the users, security roles, security policies, and credentials used to provide security services to Unwired Platform. A security provider is part of a security configuration.

**server connection** – the connection between Unwired WorkSpace and a back-end EIS.

**server farm** – the Relay Server designation for a cluster. *See cluster.*

**server-initiated synchronization** – *See push synchronization.shared read operation*

**shared read operation** – an EIS operation that returns data that the mobile application model maps into multiple MBOs. A shared read operation can load or refresh cached data for multiple MBOs.

**Simple Object Access Protocol (SOAP)** – an XML-based protocol that enables applications to exchange information over HTTP. SOAP is used when Unwired Server communicates with a Web service.

**single sign-on (SSO)** – a credential-based authentication mechanism.

**software component version (SWCV)** – a shipment unit for design-time objects in the DOE repository.

**solution** – in Visual Studio, the high-level local workspace that contains the projects users create.

**Solution Explorer** – in Visual Studio, the pane that shows the active projects in a tree view.

**Start Page** – in Visual Studio, the first page that appears when you launch the application.

**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. Use current or historical data to determine system availability or performance. Performance statistics are known as key performance indicators (KPIs).

**structured data** – data 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** – defines how data is transferred between a user's mobile device and Unwired Server. Subscriptions notify a device user of data changes, then these updates are pushed to the user's mobile device.

**synchronization** – synchronous data delivery 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* messaging-based synchronization.

**synchronization group** – defined in Unwired WorkSpace, a collection of MBOs that are synchronized simultaneously.

**synchronization parameter** – an MBO attribute that filters and synchronizes data between a mobile device and Unwired Server.

**synchronization phase** – for replication-based synchronization packages, an upload event (from device to the Unwired Server cache database) or a download event (from the cache database to the device).

**synchronize** – the process by which data consistency and population is achieved between remote disconnected clients and Unwired Server. *See* data refresh.truststore

**System Landscape Directory (SLD)** – a central repository of system landscape information used to manage the software life cycle when Sybase Unwired Platform is deployed as part of a larger SAP landscape.

**Transmit** – The term that describes the initiation of the synchronization process between the Agentry Client and Agentry Server.

**truststore** – the location in which certificate authority (CA) signing certificates are stored. *See also* keystore.

**undeploy** – remove a domain package from an Unwired Server.

**user** – Unwired Platform 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 Unwired Platform 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** – an integrated development environment product that you can use to develop device applications from generated Unwired WorkSpace code.

**Web Service Definition Language (WSDL) file** – 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.

**Welcome page** – in Eclipse, the first set of pages that appear when you launch the application.

**workspace** – in Eclipse, 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.

# Index

.NET 62

## A

administration performance 88  
 administration perspective  
   empty SCC console screen 89  
   gray SCC console screen 89  
   rectangular box instead of SCC console 89  
 administrators  
   login accepted when authentication provider  
     unavailable 94  
 Advantage Database Server  
   authentication error 83  
   service 83  
 agent.log file 90  
 application management issues 107  
 application user management issues 107  
 authentication  
   provider unavailable but administrator can log  
     in 94  
 AuthorizationException 105  
 auto purge  
   removing domain log data 98  
   removing monitor data 98

## C

cache database  
   troubleshooting 71  
 cache database problems 67  
 caching of login credentials 104  
 CallbackHandler 37, 41  
 cannot access Application Connections 57  
 cannot access sampledb  
   Enterprise Server (production) license 58,  
     111  
   Personal Development license 58, 111  
 cannot generate code with Javadoc option enabled  
   122  
 cannot insert DEFAULT in Timestamp column 129  
 cannot start Unwired Server 100  
 certificate problems 93  
 changing host name 49  
 cleaning up the Flash Player cache 95

Client Failed to Initialize error on Android device  
   168

cluster  
   primary server shutdown 84  
 cluster problems 83  
 code generation fails 120  
 command line utilities, failures of 169  
 connection errors 103  
 Connection Failed 84  
 connection pools  
   configuring 71  
 connections  
   troubleshooting 66, 71  
 credentials  
   old, ability to authenticate with 104

## D

data change notification message  
   workflow DCN 86  
 data services do not start 85  
 data source  
   connections fail 112  
   troubleshooting 66  
 data tier  
   authentication error in cluster 83  
   JVM heap space 78, 84  
 debugging 32, 35, 37, 41, 136  
   SAP DOE-C process flow 44  
   SAP JCo process flow 46  
 DEFAULT in Timestamp column 129  
 degrading performance 105  
 deploy failure for large packages 106  
 deploying the Hybrid App 142  
 deployment issues for packages 106  
 deployment problems 63  
 development edition 112  
 device log error 107  
 device provisioning problems 153  
 device user name not displayed 108  
 djc-setenv.bat 169  
 DNS server failure 91  
 documentation roadmap 3  
 DOE-C  
   connection test errors 71  
   invalid user 107

## Index

- SAP package fails 71
- DOE-C process flow 44
- domain log data
  - not displayed 98
- domain logs 7
- domain version mismatch 85

## E

- EIS connection problems 70
- ensuring system recoverability 16
- error messages
  - described 12
  - mobile business objects 171
  - mobile workflow applications 171, 218
  - severity levels 12
  - variables 12
- errors
  - user account failure 107
- exception:java.lang.Error 160
- export failure for large packages 106

## F

- F5 (browser refresh)
  - logging out of Sybase Control Center 93
- failover clusters
  - data tier authentication error 83
- Flash Player
  - cleaning up the cache after upgrade 95

## G

- gateway.log file 101
- glossaries
  - Sybase Unwired Platform terms 227
- graphic image causes OutOfMemory error 162

## H

- heap space 78, 84
- History tab is blank 96, 97
- host name changes not reflected in SCC 94
- host name resolution failure 91
- hosted Relay Server problems 81, 83

## I

- IIS application pooling 81

- intrusion detection/prevention software 56, 60
- invalid login 92

## J

- JCo process flow 46
- JMSBridge 37, 41
- JVM heap space 78, 84

## L

- LDAP
  - processes 92
  - startup 92
- listener startup errors 103
- log files
  - agent.log file 90
  - described 5
  - location 9
  - scc\_repository.log 90
- logging in to Sybase Control Center
  - clearing authentication parameters 94
- logging out of Sybase Control Center
  - unintentionally, using F5 browser refresh 93
- login invalid 92
- login troubleshooting
  - Sybase Control Center 94
- logs
  - domain-level 7
  - Unwired WorkSpace 8

## M

- maintaining host names
  - changing host name 49
- management console unavailable 94
- management issues for packages 106
- maxPoolSize 71
- maxWaitTime 71
- MBOLogger 32, 35, 37, 41
- messaging
  - connection test errors 71
- messaging installer errors 62
- messaging queue counts 105
- Messaging Synchronization Server
  - problems 80
- Microsoft Cluster issues 95
- MincheckAffectedRowCountException 144
- mlsrv12.exe process 77



mobile business objects  
 error messages 171  
 relationship guidelines 203  
 relationship restrictions 203  
 validation rules 171

mobile workflow applications  
 error messages 171, 218  
 validation rules 171, 218

monitoring  
 issues for platform components 96

monitoring data  
 not displayed 98

**N**

NetworkStreamParams property 155

**O**

OBMO.exe process 77  
 org.omg.CORBA.COMM\_FAILURE WARN 84  
 OutOfMemory error on BlackBerry simulator 162

**P**

package deployment and management issues 106  
 package type mismatch 166  
 packet dropped error 157  
 passwords  
 old, ability to authenticate with 104  
 performance  
 Sybase Control Center 95  
 performance data  
 administration 88  
 performance degradation 105  
 platform component monitoring issues 96  
 port conflict  
 cannot access Application Connections 57  
 port conflicts  
 among multiple SCC versions 91  
 with Sybase Control Center X.X 103  
 port numbers 51, 55, 57, 92  
 problems starting Sybase Control Center services  
 90  
 problems with application and application user  
 management 107

**Q**

queue counts 105

QueueNotReadyException 84

## R

reinstalling Sybase Control Center service 91

relationships  
 guidelines 203  
 restrictions 203

Relay Server  
 problems 81

Relay Server problems 83

remote server environment  
 server error 154  
 synchronization failure 154

restarting a remote server  
 unsuccessful 101

RSOE  
 startup problems 60

runtime problems 66

## S

sampledb configuration error 154

SAP 85  
 debugging DOE-C process flow 44  
 debugging JCo process flow 46  
 DOE-C connection test errors 71  
 messaging connection test errors 71  
 user account error 107

SCC console tree is not complete 105

scc\_repository.log file 90

security error when connecting to SCC 94

security problems 68

server problems 73

Server Response Timeout error on BlackBerry  
 device 163

Server Response Timeout error on BlackBerry  
 simulator 163

server tier administration issues 98

servers  
 problems 73, 80, 81

severity levels in error messages 12

showUrlInBrowser 161

SSL certificates  
 error when missing 94

starting a remote server  
 unsuccessful 101

startup problems  
 RSOE 60

## Index

- Unwired Server 60
- stopping a remote server
  - unsuccessful 101
- stream parameters 155
- SUPBridge 37, 41
- Sybase Control Center
  - cannot ping Unwired Server 76
  - failure to start 91
  - logging out unintentionally with F5 93
  - management tier issues 89
  - reinstalling the service 91
  - second version fails to start 91
  - security error when connecting 94
  - service port conflicts 103
  - Unwired WorkSpace 76
  - Windows service fails to start 90
- Sybase Control Center performance 95
- Sybase Control Center service 92
- Sybase Hosted Relay Server problems 81, 83
- synchronization fails 154
- synchronization keys 160
- synchronization problems 65, 101, 155
- system messages 12
- system recoverability 16

### T

- TCP/IP filtering causing errors 103
- terms
  - Sybase Unwired Platform 227
- testing database file on SD card from Windows
  - Mobile simulator 166
- time-change related problems for Unwired Server 75
- troubleshooting
  - basic scenarios 17
  - basics 5
  - cache 26
  - cache database problems 67
  - changing host name 49
  - cluster problems 83
  - command line utility problems 169
  - data change notification 28
  - deployment problems 63
  - device client problems 153
  - device provisioning 153
  - Eclipse mobile business object development 111
  - EIS connection problems 70
  - end-to-end process flow 17

- Hybrid App development 138
- Hybrid Web Container 133
- message synchronization 20, 133
- post-installation configuration 49
- Relay Server problems 81
- replication synchronization 23, 125
- runtime problems 66
- security problems 68
- server problems 73
- Sybase Control Center problems 87
- Sybase Hosted Relay Server 83
- synchronization problems 65
- system level problems 49
- system performance issues 73
- Unwired Server problems 73, 80, 101
- Unwired Server startup 100
  - user account failure 107
- troubleshooting Microsoft Cluster 95
- troubleshooting performance issues 88
- troubleshooting Unwired Platform with SCC 87

### U

- Unwired Platform management console unavailable 94
- Unwired Server
  - cannot ping from Sybase Control Center 76
  - cannot ping from Unwired WorkSpace 76
  - configuration changes unsuccessful 103
  - extended session 92
  - list does not appear in SCC 99
  - problems 73
  - process of stopping service hangs 75
  - refresh after changing configuration 103
  - startup failure 100
  - startup failure after time change 75
  - startup problems 60
  - Unwired Server service 77
- Unwired WorkSpace logs 8
- users
  - able to connect with old password 104
  - not displayed for registered devices 108

### V

- validation rules
  - mobile business objects 171
  - mobile workflow applications 171, 218
- variables in error messages 12

viewing assigned port numbers 55

## **W**

Windows application event log 7

Windows Mobile devices

cannot test database file on SD card from  
simulator 166

WorkflowClient log 142

