SYBASE®

Installation and Configuration Guide

Sybase Mobile Sales for SAP®

CRM 1.0

Contents

Contents

CHAPTER 1 Sybase Mobile Sales for SAP CRM Solution Architecture

The Sybase® Mobile Sales for SAP® CRM solution enables the availability of key SAP CRM processes on iPhone and Windows Mobile devices. The architecture leverages components from both SAP and Sybase to deliver a comprehensive push-enabled end-to-end solution.

The key components of the architecture include:

- SAP® Customer Relationship Management (CRM)
- SAP NetWeaver® Mobile – this SAP middleware component provides for data consolidation and data distribution services.
- Sybase Unwired Platform – Sybase Unwired Platform delivers the Sybase SAP DOE Connector that receives push messages from SAP and guarantees delivery to mobile devices.
- Sybase Relay Server – Relay Server solves IT security problems by letting servers and devices connect without requiring changes to your existing corporate firewall configurations and policies.
- Sybase Mobile Sales for SAP CRM application, which runs on iPhone and Windows Mobile devices.

Sybase Mobile Sales for SAP CRM Solution Architecture
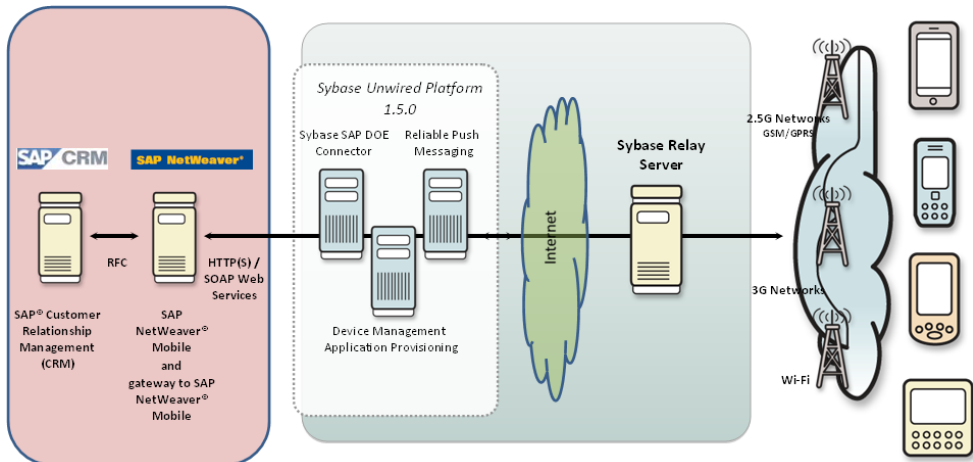
CHAPTER 2   **Installation Workflow for Windows Mobile**

This workflow defines a path for system administrators to install and provision the Sybase Mobile Sales for SAP® CRM application to a Windows Mobile device.

Note: For information on customizing the application for your organization, see the *Sybase Mobile Sales for SAP Customization White Paper*.

Installation Tasks

1. Review system and device requirements. See *Chapter 4, Planning Your Installation* on page 9.
2. Install Sybase Unwired Platform Deployment Edition. See the Installation Workflows in the *Sybase Unwired Platform Installation Guide*.
3. Install Sybase SAP® DOE Connector. See the Installation Workflows in the *Sybase SAP DOE Connector Installation and Configuration Guide*.
4. Install Sybase Mobile Sales for SAP CRM using the installation media.
   The installer installs the CAB files Windows Mobile Professional and Windows Mobile Standard.
   - `MoMessaging_ppc.cab` for Windows Mobile Professional, or `MoMessaging_sp.cab` for Windows Mobile Standard
   - `SUPObj_ppc.cab` for Windows Mobile Professional, or `SUPObj_sp.cab` for Windows Mobile Standard
   - `SybaseMobileSales_Pro_<version>.CAB` for Windows Mobile Professional, or `SybaseMobileSales_Std_<version>.CAB` for Windows Mobile Standard

   The CAB files install in the `Mobile Sales for SAP CRM` subdirectory under the Sybase Unwired Platform installation directory. Install the Mobile Sales application on the same machine as you installed Sybase Unwired Platform and Sybase SAP DOE Connector.

Provisioning Tasks

1. Register each device using Sybase Control Center so it can connect to Unwired Server. For each device, be sure to note the user name, Unwired Server name, Unwired Server port number, relay server farm ID (if applicable), and activation code.
   See *Registering and Setting Up Messaging Devices* in the Sybase Control Center online help.

2. Ensure all device users have an SAP account.
3. (Optional) Modify the Mobile Sales system configuration for Windows Mobile in the `SybaseCRM.Configuration.xml` file. You must update and provision this file to the device before users log into the application (subscribe) for the first time.
   See .*Appendix B, Mobile Sales Configuration File for Windows Mobile* on page 37.
4. Provision the application files to the device.

How many devices are you provision-ing?	Method	Where to find more information
Personal/developer in-stallations	Copy the CAB file to each device to install.	*Sybase Mobile Sales for SAP CRM User Guide for Windows Mobile > Installing Mobile Sales on Your Windows Mobile Device.*
Enterprise/large installa-tions	Use Sybase Unwired Platform to provision the application to multiple devices.	*Sybase Unwired Platform System Ad-ministration > Systems Administration > Device User Management > Device Pro-visioning > Afaria Provisioning and De-vice Management.*

User Tasks

1. Device users install device prerequisites. See *Installation Prerequisities* on page 11.
2. Ensure that device users create a Notifications messaging account. See *Sybase Mobile Sales for SAP CRM User Guide for Windows Mobile > Creating a Notifications Messaging Folder*.
3. Per the system administrator's instructions, users install the CAB files on the device.
4. After CAB file installation, users indicate the Unwired Server connection information in Sybase Settings before launching the Mobile Sales application.
   See *Sybase Mobile Sales for SAP CRM User Guide for Windows Mobile > Installing Mobile Sales on Your Windows Mobile Device.*
5. Device users start Sybase Mobile Sales for SAP CRM.
   See *Sybase Mobile Sales for SAP CRM User Guide for Windows Mobile > Starting Sybase Mobile Sales for SAP CRM.*
6. If necessary, users can unsubscribe or resubscribe from the device.
   See *Sybase Mobile Sales for SAP CRM User Guide for Windows Mobile > Unsubscribing and Resubscribing to Mobile Sales.*

See also

- *Chapter 5, Installing Mobile Sales for Windows Mobile Using the Installer* on page 13

CHAPTER 3   **Installation Workflow for iPhone**

This workflow defines a path for system administrators to install and provision the Sybase
Mobile Sales for SAP CRM application to the iPhone device.

Installation Tasks

1. Review system and device requirements. See *Chapter 4, Planning Your Installation* on
   page 9.
2. Install Sybase Unwired Platform Deployment Edition. See the Installation Workflows in
   the *Sybase Unwired Platform Installation Guide*.
3. Install Sybase SAP DOE Connector. See the Installation Workflows in the *Sybase SAP
   DOE Connector Installation and Configuration Guide*.

Provisioning Tasks

1. Register each device using Sybase Control Center so it can connect to Unwired Server. For
   each device, be sure to note the user name, Unwired Server name, Unwired Server port
   number, relay server farm ID (if applicable), and activation code.
   See *Registering and Setting Up Messaging Devices* in the Sybase Control Center online
   help.
2. Ensure all device users have an SAP account.
3. Provision the application files to the device and install the application.

Are you customizing the application?	Method	Where to find more information
No	Download and install the Sybase Mobile Sales & Workflow application from the App Store.	*Sybase Mobile Sales for SAP CRM User Guide for iPhone > Installing Mobile Sales on Your iPhone Device > Installing Mobile Sales from App Store*
Yes	Provision the application profile and application to the device. You can also place the application image on the network and send users information about how to obtain it and install it using iTunes.	• *Sybase Unwired Platform System Administration > Systems Administration > Device User Management > Device Provisioning > Apple Provisioning for iPhone* • *Sybase Mobile Sales for SAP CRM User Guide for iPhone > Installing Mobile Sales on Your iPhone Device > Installing Mobile Sales Using iTunes*

User Tasks

1. Device users indicate the Unwired Server connection information in Sybase Settings before launching the Mobile Sales application. If you do not enter the connection information, the application prompts you to enter this information the first time you start it.

   See *Sybase Mobile Sales for SAP CRM User Guide for iPhone > Installing Mobile Sales on Your iPhone Device > Connection Settings*. User can also run the Mobile Sales application in demo mode, which does not require Unwired Server connection information.

2. Users start Sybase Mobile Sales for SAP CRM.

   See *Sybase Mobile Sales for SAP CRM User Guide for iPhone > Starting Sybase Mobile Sales for SAP CRM*.

3. If necessary, users can unsubscribe or resubscribe from the device.

   See *Sybase Mobile Sales for SAP CRM User Guide for iPhone > Unsubscribing and Resubscribing to Mobile Sales*.

CHAPTER 4   **Planning Your Installation**

Before you install or upgrade, plan your environment. View the release bulletin for the latest information on system requirements and patches.

1. Make sure your system meets all the requirements for your installation scenario.
2. Install the prerequisites.

See also
• *Chapter 5, Installing Mobile Sales for Windows Mobile Using the Installer* on page 13

System Requirements

Make sure your system meets all requirements before you install. See the release bulletin for the most current information about required system patches.

• For system and disk space requirements for Sybase Unwired Platform, including Unwired Server, Relay Server, and Afaria, see the *Sybase Unwired Platform Installation Guide*.
• For system and disk space requirements for Sybase SAP DOE Connector, see the *Sybase SAP DOE Connector Installation and Configuration Guide*.

This table indicates the disk space requirements when installing Sybase Mobile Sales for SAP CRM on the machine running Sybase Unwired Platform using the Mobile Sales installation media.

Table 1. Disk space requirements

Product	Supported
Sybase Mobile Sales for SAP CRM	99.0 MB

Table 2. SAP Compatibility Matrix

SAP product	Supported versions
SAP® Customer Relationship Management (CRM)	2007 SP04 and later 7.0 SP03 and later
SAP NetWeaver® Mobile	7.1 SP07 and later
gateway to SAP NetWeaver® Mobile	1.1 and later

SAP product	Supported versions
Entity Set Definitions for Mobile Applications (ESDMA)	SP01

See also
- *Installation Prerequisities* on page 11

Device Requirements

Sybase Mobile Sales for SAP CRM can run on these devices.

Supported Devices

- Windows Mobile 6.0.x Standard
- Windows Mobile 6.1.x Standard
- Windows Mobile 6.0.x Professional
- Windows Mobile 6.1.x Professional
- Apple iPhone 3G, software version 3.1.x
- Apple iPhone 3GS, software version 3.1.x
- Apple iPod Touch 3G, software version 3.1.x

Note: For better performance, Sybase recommends using Apple iPhone 3GS.

Supported Languages

- English
- French
- German

Device Storage Requirements

Make sure you have the minimum free storage space before installing the Mobile Sales application on the device. You may need more depending on how much data you store on the device.

- Windows Mobile: 40MB internal storage
- iPhone: 5MB

See also
- *Installation Prerequisities* on page 11

Installation Prerequisities

Install the required prerequisites for Sybase Mobile Sales for SAP CRM before launching the installer.

Install Mobile Sales on the same machine as Sybase Unwired Platform and Sybase SAP DOE Connector.

Table 3. Server prerequisites

Product	Version	Description
Sybase Unwired Platform, Deployment Edition	1.5	First, install Sybase Unwired Platform from the Sybase Unwired Platform installation media.
Sybase SAP DOE Connector	1.0	Second, install Sybase SAP DOE Connector from the Sybase SAP DOE Connector installation media.

Table 4. Windows Mobile device prerequisites

Product	Version	Description
Microsoft ActiveSync	4.5 or later	(Optional) If provisioning the files manually, allows synchronization of files between the Sybase Unwired Platform machine and the Windows Mobile emulator or device. Download it on the Microsoft web site at *http://www.microsoft.com/windowsmobile/en-us/downloads/microsoft/activesync-download.mspx*.
.NET Compact Framework	3.5	You can download it from the Microsoft Web site at *http://www.microsoft.com/downloads/details.aspx?FamilyID=E3821449-3C6B-42F1-9FD9-0041345B3385&displaylang=en*.
.NET Compact Framework Messages for Windows Mobile	3.5	(Optional) Provides localized Windows Mobile error messages. All Mobile Sales messages are available in the supported languages.

See also
- *Device Requirements* on page 10

- *System Requirements* on page 9

CHAPTER 5   **Installing Mobile Sales for Windows Mobile Using the Installer**

Install Sybase Mobile Sales for SAP CRM for Windows Mobile from the installation media on the same machine in which you installed Sybase Unwired Platform and Sybase SAP DOE Connector.

Prerequisites

- Install the prerequisites.
- Shut down any Sybase products or processes, and associated third-party processes.
- Verify that you are logged in with administrtive rights.
- Uninistall any existing version of Sybase Mobile CRM for SAP.

Sybase Mobile Sales for SAP CRM installs in the Sybase Unwired Platform installation directory, by default, `C:\Sybase\UnwiredPlatform-1_5\Mobile Sales for SAP CRM`.

1. Insert the Sybase Mobile Sales for SAP CRM installation media.

   If you are installing from a Web download, extract the archive to a temporary directory on the local disk, not a shared drive.
2. On the installer welcome page, click **Next**.
3. Accept the terms of the license agreement.
   a) Select the appropriate geographic location and read the license agreement.
   b) Select **I Do Agree to the Terms of the Sybase License, for the Install Location Specified**, then click **Next**.
4. Click **Install**.

Next
Provision the Mobile Sales application to the device.

See also
- *Chapter 4, Planning Your Installation* on page 9
- *Chapter 2, Installation Workflow for Windows Mobile* on page 3

CHAPTER 6   **Uninstalling Mobile Sales for Windows Mobile**

If you installed Sybase Mobile Sales for SAP CRM for Windows Mobile using the installation media, you can uninstall the application using the Add or Remove Programs option on the Control Panel, which launches the uninstaller. Uninstall Mobile Sales for Windows Mobile before reinstalling or upgrading.

Prerequisites
Shut down any Sybase products or processes.

The uninstaller uninstalls the Windows Mobile CAB files from the Sybase Unwired Platform installation directory.

Note: To uninstall the application from a Windows Mobile or an iPhone device, see the *Sybase Mobile Sales for SAP CRM User Guide for Windows Mobile* or *Sybase Mobile Sales for SAP CRM User Guide for iPhone*, respectively.

1. From the Control Panel, select **Add or Remove Programs**.
2. Select **Sybase Mobile Sales for SAP CRM 1.0**, and click **Change/Remove**.
3. Click **Next** in the welcome window.
4. Click **Uninstall**.
5. Click **Finish** when the following message appears: `The InstallShield wizard has successfully uninstalled Sybase Mobile Sales for SAP CRM 1.0.`

Sybase Mobile Sales for SAP CRM

CHAPTER 7   **Troubleshooting for Windows Mobile**

Provides troubleshooting information when installing and configuring Sybase Mobile Sales for SAP CRM for Windows Mobile. See the Sybase Mobile Sales for SAP CRM Release Bulletin for the latest information. For information on Sybase Unwired Platform issues, see the Sybase Unwired Platform Release Bulletin.

Wrong Device for Code

Issue: When entering the Unwired Server connection information, the error `Wrong Device for Code` appears in the Sybase Settings log.

Solution:

1. Use Sybase Control Center to delete the user.
2. Clone the user.
3. Delete the old user.
4. On the device, select **Retry**, then **Done**.
5. Try again to connect.

Device Is Not Connected to Unwired Server

Issue: Device must be cradled or have network connectivity.

Solution 1: Check that the device is connected to Unwired Server:

- **Start > Programs > Sybase Settings**.
- Select **Menu > Show Log**.
- Check the log for `Connected to Server`, and look at the timestamp to see if you are currently connected.

Solution 2: Check that device is connected from Unwired Server:

- Start Sybase Control Center. See *Sybase Unwired Platform Installation Guide*.
- Check if the status is connected and a device value is present. See *Checking Unwired Server Status* in Sybase Control Center online help.

CHAPTER 8 **Troubleshooting for iPhone**

Provides troubleshooting information when installing and configuring Sybase Mobile Sales for SAP CRM for iPhone. See the Sybase Mobile Sales for SAP CRM Release Bulletin for the latest information. For information on Sybase Unwired Platform issues, see the Sybase Unwired Platform Release Bulletin.

Connection Fails With No Error Message

Issue: Mobile Sales for iPhone does not always provide connection failed errors on the device.

Workaround: If you suspect that you are not connected to the SAP system, for example, if you did not enter the Unwired Server connection information correctly, you can check on the device or by using Sybase Control Center.

On the device:

* From within the Mobile Sales application, tap **Settings > General > Connection Information**.
* View the **Connection** status: Failed or Active.

On Sybase Control Center:

* Start Sybase Control Center. See *Sybase Unwired Platform Installation Guide*.
* For the Unwired Platform Cluster, select **Users > Messaging**. Click **Get Trace** to retrieve the device log. See also Sybase Control Center online help.

APPENDIX A **Sybase SAP DOE Connector Command Line Utility Reference**

The Sybase SAP DOE Connector Command Line Utility provides commands to manage ESDMA packages and subscriptions to those packages.

The Sybase SAP DOE Connector Command Line Utility is a text-based console that you run from the command line. You can work interactively in the console, or you can write batch files that silently execute any sequence of commands.

The commands that are available in the Command Line Utility allow you to manage ESDMA packages and subscriptions to those packages. You must use the **deploy** command to deploy a package.

The functionality of the other commands is available through Sybase Control Center. See *Sybase Control Center > Configure > Configuring Unwired Platform > Packages*.

Managing the Console

Use the administrative commands to start the Command Line Utility console, log in, get help, and exit

Starting the Command Line Utility Console

Before you can use the DOE-C Command Line Utility interactively, you must start the console.

1. In Windows Explorer or at a command prompt, navigate to `%DOE-C_CLU_HOME%/bin/`.
2. Start up `clu.bat`.
3. Log in, or enter commands without a login.

   If you enter a command without logging in first, you are prompted to enter the DOE-C server admin listener URL, user name, and password for the first command that you enter. This does not apply to help and exit. After entering this information for your first command, you are not prompted when you enter additional commands.

login Command

Use the login command to log in to the DOE-C Command Line Utility console.

If you do not use the **login** command to log in to the Command Line Utility console, you are prompted to enter the login information for each command that you enter.

Syntax

```
login [-u|--DOECServerUser SCCUuser] [-pw|--password
SCCUUserPassword]
[-url|--DOECSocketListenerUrl Url] [-h|--help] [-sl|--silent]
```

Parameters

- **-h|--help** – Gets help on this command.
- **-u|--DOEServerUser** – Specifies the Sybase Control Center admin user account.
- **-pw|--password** – Specifies the Sybase Control Center admin user account password.
- **-url|--DOECSocketListenerUrl** – Specifies the DOE server socket listener URL. For example, iiop://myhost:2000.
- **-sl|--silent** – Disables all user interactive questions; this option is mainly used when writing a batch file.

exit Command

Use the exit command to close the DOE-C Command Line Utility console.

The **exit** command closes the Command Line Utility console.

Syntax

```
exit [-h|--help]  [-sl|--silent]
```

Parameters

- **-h|--help** – Gets help on this command.
- **-sl|--silent** – Disables all user interactive questions; this option is mainly used when writing a batch file.

help Command

Use the help command to get help on DOE-C Command Line Utility commands.

The **help** command displays help text for any specific DOE-C command, or for all commands.

Syntax

```
help commandName | [-a|--all]  | [-h|--help]
```

Parameters

- **-h|--help** – Gets help on the **help** command.
- **-a|--all** – Gets help on all commands.
- *commandName* – Gets help on the specified command.

Aborting Commands

You can abort commands in interactive mode.

When you are in interactive mode, there are two ways to abort commands.

Abort method	Description
Press Ctrl+C at any time	Aborts command in progress and exits from Command Line Utility console. • Works at any time with any command. • To enter additional commands after using this option, you must restart the console.
Enter "abort" when prompted	Aborts command in progress without exiting from Command Line Utility console. • Only available with certain commands. • Only available when prompted. • After using this option, the console is ready for you to continue entering commands.

Managing Packages

The DOE-C Command Line Utility provides commands to deploy and configure packages from the command line.

deploy Command

Use the deploy command to deploy a DOE-C package to Sybase Unwired Server.

You must use the **deploy** command in the DOE-C Command Line Utility to deploy a DOE-C package. This is the only DOE-C Command Line Utility command that is not available in the Sybase Control Center in Sybase Unwired Platform version 1.5.

Syntax

```
deploy [-dir|--deployFilesDirectory deploymentDirectory]
[-u|--technicalUser SAPUserAccount]
[-pw|--password SAPUserPassword] [-h|--help] [-sl|--silent]
```

Parameters

• **-h|--help** – Gets help on this command.

- **-dir|--deployFilesDirectory** – Specifies the directory location that contains deployment files.
- **-u|--technicalUser** – Specifies the SAP technical user account to use when sending non-client-based requests.
- **-pw|--password** – Specifies the SAP technical user account password.
- **-sl|--silent** – Disables all user interactive questions; this option is mainly used when writing a batch file.

getPackages Command

Use the getPackages command to get a list of deployed DOE-C packages, or to return detailed information for one or more specified packages.

Syntax

```
getPackages [-i|in inputXmlFile] [-o|out outputXmlFile]
[-ps|--packageNames nameAndVersionList] [-h|--help] [-sl|--silent]
```

Parameters

- **-h|--help** – Gets help on this command.
- **-i|--in** – Reads package name from input XML file. You can generate the XML file by using the -o parameter with the getPackages command.
- **-o|--out** – Saves command output to an XML file.
- **-ps|--packageNames** – Specifies one or more package names for which detailed information will be returned. Each package name is followed by a colon and the package version number. Use a comma to separate the information for multiple packages, with no white space; for example:

```
-ps myPkg1:2.0,myPkg2:1.0
```

Note: If you omit this parameter, the getPackages command returns a list of deployed packages.

- **-sl|--silent** – Disables all user interactive questions; this option is mainly used when writing a batch file.

setEndpointProperties Command

Use the setEndpointProperties command to set the DOE endpoint properties for a deployed DOE-C package.

You must set the SAP technical user account ID and password so that DOE-C can communicate with the SAP server. You may also want to set the HTTP timeout value to use when sending messages to the SAP DOE. You can use the setEndpointProperties Command to set all of these DOE endpoint properties.

Syntax

```
setEndpointProperties [-i|in inputXmlFile] [-o|out outputXmlFile]
[[-a|--all] | [-p|--packageName name]]
[-ds|--doePacketDropSize byteSize]
[-u|--technicalUser SAPUserAccount]
[-pw|--password SAPUserPassword] [-t|--httpTimeout seconds]
[-h|--help] [-sl|--silent]
```

Parameters

- **-h|--help** – Gets help on this command.
- **-i|--in** – Reads package name from input XML file. You can generate the XML file by using the -o parameter with the getPackages command.
- **-o|--out** – Saves command output to an XML file.
- **-a|--all** – Specifies that endpoint properties will be set for all deployed packages.
- **-p|--packageName** – Specifies package name for which endpoint properties will be set. Package name is followed by a colon and the package version number, with no white space; for example:

  ```
  -p myPkg:2.0
  ```
- **-ds|--doePacketDropSize** – Specifies the DOE packet drop size, in bytes. This is the size of the largest JSON message the DOE Connector will process on behalf of a JSON client; default is one megabyte. DOE messages larger than the DOE packet drop size, other than asynchronous search responses, place the subscription in the packet dropped state.
- **-u|--technicalUser** – Specifies the SAP technical user account to use when sending non-client-based requests.
- **-pw|--password** – Specifies the SAP technical user account password.
- **-t|--httpTimeout** – Specifies the HTTP timeout value, in seconds, to use when sending messages to the SAP DOE.
- **-sl|--silent** – Disables all user interactive questions; this option is mainly used when writing a batch file.

getEndpointProperties Command

Use the getEndpointProperties command to get the DOE endpoint properties (HTTPTimeout value) for a deployed DOE-C package.

Syntax

```
getEndpointProperties [-i|in inputXmlFile]
[[-a|--all] | [-ps|--packageNames name]]
[-h|--help] [-sl|--silent]
```

Parameters

- **-h|--help** – Gets help on this command.

- **-i|--in –** Reads command input from an XML file.
- **-a|--all –** Specifies that endpoint properties will be returned for all deployed packages.
- **-ps|--packageNames –** Specifies one or more package names for which endpoint properties will be returned. Each package name is followed by a colon and the package version number. Use a comma to separate the information for multiple packages, with no white space; for example:

```
-ps myPkg1:2.0,myPkg2:1.0
```

- **-sl|--silent –** Disables all user interactive questions; this option is mainly used when writing a batch file.

setPackageLogLevel Command

Use the setPackageLogLevel command to set the log level for one or more deployed DOE-C packages.

The log level that you set for a deployed DOE-C determines the amount of information that is logged for that package.

Syntax

```
setPackageLogLevel [-i|in inputXmlFile] [-l|logLevel level]
[[-a|--all] | [-ps|--packageNames name]] [-h|--help] [-sl|--silent]
```

Parameters

- **-h|--help –** Gets help on this command.
- **-i|--in –** Reads package name from input XML file. You can generate the XML file by using the -o parameter with the getPackages command.
- **-l|--logLevel –** Specifies the log level to be set, from the following values:

  - **DEBUG** - provides the maximum amount of detail that can be logged.
  - **INFO** - adds informational messages to information logged by WARN.
  - **WARN** - adds less serious warnings to information logged by ERROR
  - **ERROR** - only error messages are logged.
  - **OFF** - no information is logged.
- **-a|--all –** Specifies that log level will be set will be set for all deployed packages.
- **-ps|--packageNames –** Specifies one or more package names for which log level will be set. Each package name is followed by a colon and the package version number. Use a comma to separate the information for multiple packages, with no white space; for example:

```
-p myPkg:2.0
```

- **-sl|--silent –** Disables all user interactive questions; this option is mainly used when writing a batch file.

getPackageLogLevel Command

Use the getPackageLogLevel command to get the log level for one or more deployed DOE-C packages.

Syntax

```
getPackageLogLevel [-i|in inputXmlFile] [-o|out outputXmlFile]
[-ps|--packageNames name] [-h|--help] [-sl|--silent]
```

Parameters

- **-h|--help** – Gets help on this command.
- **-i|--in** – Reads package name from input XML file. You can generate the XML file by using the -o parameter with the getPackages command.
- **-ps|--packageNames** – Specifies one or more package names for which detailed information will be returned. Each package name is followed by a colon and the package version number. Use a comma to separate the information for multiple packages, with no white space; for example:

```
-p myPkg:2.0
```

  Note: If -ps is omitted, the getPackageLogLevel command returns a list of log levels for all deployed packages.

- **-sl|--silent** – Disables all user interactive questions; this option is mainly used when writing a batch file.

removePackages Command

Use the removePackages command to remove one or more deployed DOE-C packages from the Sybase Unwired Server.

The removePackages command reverses the action of the deploy command.

Syntax

```
removePackages [-i|in inputXmlFile] [-ps|--packageNames name]
[-h|--help] [-sl|--silent]
```

Parameters

- **-h|--help** – Gets help on this command.
- **-i|--in** – Reads package name from input XML file. You can generate the XML file by using the -o parameter with the getPackages command.
- **-ps|--packageNames** – Specifies one or more package names which are to be removed. Each package name is followed by a colon and the package version number. Use a comma to separate the information for multiple packages, with no white space; for example:

```
-p myPkg:2.0
```

Note: If -ps is omitted, the removePackages command prompts interactively for package names to be removed.

- **-sl|--silent –** Disables all user interactive questions; this option is mainly used when writing a batch file.

Managing Package Subscriptions

The DOE-C Command Line Utility provides commands to deploy and configure packages from the command line.

getSubscriptions Command

Use the getSubscriptions command to get information on subscriptions to one or more deployed DOE-C packages.

Syntax

```
getSubscriptions [-i|in inputXmlFile] [-o|out outputXmlFile]
[-f|--filter filterExpression] [-ps|--packageNames name]
[-h|--help] [-sl|--silent]
```

Parameters

- **-h|--help –** Gets help on this command.
- **-i|--in –** Reads package name from input XML file. You can generate the XML file by using the -o parameter with the getPackages command.
- **-o|--out –** Saves command output to an XML file.
- **-f|--filter –** Specifies the filter to use on the subscriptions. Each column name is followed by a colon and the filter string. Use a comma to separate the information for multiple column names, with no white space; for example:

  ```
  -f columnName:filterString,columcName2:filterString2
  ```

  Valid filter column names are: subscriptionID, packageName, clientID, physicalID, logicalID, userName, language, clientMsgID,clientMsgTimeStamp, serverMsgID, serverMsgTimeStamp, logLevel.

  You can use "?" and "*" wildcard characters in your filter strings; for example:

  ```
  -f clientMsgTimeStamp:*Jan*21?41*2009,userName:john*
  ```

- **-ps|--packageNames –** Specifies one or more package names for which subscription information will be returned. Each package name is followed by a colon and the package version number. Use a comma to separate the information for multiple packages, with no white space; for example:

  ```
  -p myPkg:2.0
  ```

Note: If -ps is omitted, the getSubscriptions command prompts you for a package name.

- **-sl|--silent –** Disables all user interactive questions; this option is mainly used when writing a batch file.

setSubscriptionsLogLevel Command

Use the setSubscriptionsLogLevel command to set the log level for subscriptions to a deployed DOE-C package.

Syntax

```
setSubscriptionsLogLevel [-i|in inputXmlFile]
[-l|--logLevel level] [[-a|--all] | [-p|--packageName name]]
[-s|--subscriptionID ID] [-h|--help] [-sl|--silent]
```

Parameters

- **-h|--help –** Gets help on this command.
- **-i|--in –** Reads package name from input XML file. You can generate the XML file by using the -o parameter with the getPackages command.
- **-l|--logLevel –** Specifies the log level to set as one of the following:

    - **DEBUG** - provides the maximum amount of detail that can be logged.
    - **INFO** - adds informational messages to information logged by WARN.
    - **WARN** - adds less serious warnings to information logged by ERROR
    - **ERROR** - only error messages are logged.
    - **OFF** - no information is logged.
- **-a|--all –** Specifies that log level will be set for all deployed packages.
- **-p|--packageName –** Specifies package name for which log level will be set. Package name is followed by a colon and the package version number, with no white space; for example:

    ```
    -p myPkg:2.0
    ```
- **-s|--subscriptionID –** Specifies one or more subscription IDs for which you want to set the log level. Use a comma to separate multiple subscription IDs, with no white space; for example:

    ```
    -s mySubs1,mySubs2
    ```
- **-sl|--silent –** Disables all user interactive questions; this option is mainly used when writing a batch file.

getSubscriptionsLogLevel Command

Use the getSubscriptionsLogLevel command to get the log level for subscriptions to a deployed DOE-C package.

Syntax

```
getSubscriptionsLogLevel [-i|in inputXmlFile]
[-p|--packageName name] [-s|--subscriptionID ID]
[-h|--help] [-sl|--silent]
```

Parameters

- **-h|--help** – Gets help on this command.
- **-i|--in** – Reads command input from an XML file.
- **-p|--packageName** – Specifies package name for which detailed information will be returned. Package name is followed by a colon and the package version number, with no white space; for example:

  ```
  -p myPkg:2.0
  ```
- **-s|--subscriptionID** – Specifies one or more subscription IDs for which you want to get the log level. Use a comma to separate multiple subscription IDs, with no white space; for example:

  ```
  -s mySubs1,mySubs2
  ```

  Note: If -s is omitted, the getSubscriptionsLogLevel command returns a list of all subscriptions for the specified package.

- **-sl|--silent** – Disables all user interactive questions; this option is mainly used when writing a batch file.

recoverSubscriptions Command

Use the recoverSubscriptions command to recover subscriptions to a deployed DOE-C package.

Syntax

```
recoverSubscriptions [-i|in inputXmlFile]
[[-a|--all] | [-p|--packageName name]] [-s|--subscriptionID ID]
[-h|--help] [-sl|--silent]
```

Parameters

- **-h|--help** – Gets help on this command.
- **-i|--in** – Reads package name from input XML file. You can generate the XML file by using the -o parameter with the getPackages command.
- **-a|--all** – Specifies that subscriptions will be recovered for all deployed packages.

- **-p|--packageName –** Specifies package name for which subscriptions will be recovered. Package name is followed by a colon and the package version number, with no white space; for example:
  ```
  -p myPkg:2.0
  ```
- **-s|--subscriptionID –** Specifies one or more subscription IDs which you want to recover. Use a comma to separate multiple subscription IDs, with no white space; for example:
  ```
  -s mySubs1,mySubs2
  ```
- **-sl|--silent –** Disables all user interactive questions; this option is mainly used when writing a batch file.

resyncSubscriptions Command

Use the resyncSubscriptions command to reactivate subscriptions to a deployed DOE-C package.

If the status of a subscription is "suspended," the resyncSubscriptions command changes the status to "active." If the status is already "active" the resyncSubscriptions command has no effect.

If the Sybase SAP DOE Connector does not respond to the SAP DOE quickly enough, the DOE may mark that subscription's queues as "blocked" and stop sending messages to the DOE-C. At startup, the DOE-C sends a status message to the DOE that should unblock these queues. Should this happen at another time, a resyncSubscriptions command may be used to resume communication from the DOE to the DOE-C.

Syntax
```
resyncSubscriptions [-i|in inputXmlFile]
[[-a|--all] | [-p|--packageName name]] [-s|--subscriptionID ID]
[-h|--help] [-sl|--silent]
```

Parameters

- **-h|--help –** Gets help on this command.
- **-i|--in –** Reads package name from input XML file. You can generate the XML file by using the -o parameter with the getPackages command.
- **-a|--all –** Specifies that subscriptions will be reactivated for all deployed packages.
- **-p|--packageName –** Specifies package name for which subscriptions will be reactivated. Package name is followed by a colon and the package version number, with no white space; for example:
  ```
  -p myPkg:2.0
  ```
- **-s|--subscriptionID –** Specifies one or more subscription IDs which you want to recover. Use a comma to separate multiple subscription IDs, with no white space; for example:
  ```
  -s mySubs1,mySubs2
  ```
- **-sl|--silent –** Disables all user interactive questions; this option is mainly used when writing a batch file.

endSubscriptions Command

Use the endSubscriptions command to end subscriptions to a deployed DOE-C package.

Syntax

```
endSubscriptions [-i|in inputXmlFile]
[[-a|--all] | [-p|--packageName name]] [-s|--subscriptionID ID]
[-h|--help] [-sl|--silent]
```

Parameters

- **-h|--help** – Gets help on this command.
- **-i|--in** – Reads package name from input XML file. You can generate the XML file by using the -o parameter with the getPackages command.
- **-a|--all** – Specifies that subscriptions will be ended for all deployed packages.
- **-p|--packageName** – Specifies package name for which subscriptions will be ended. Package name is followed by a colon and the package version number, with no white space; for example:

  ```
  -p myPkg:2.0
  ```
- **-s|--subscriptionID** – Specifies one or more subscription IDs which you want to recover. Use a comma to separate multiple subscription IDs, with no white space; for example:

  ```
  -s mySubs1,mySubs2
  ```
- **-sl|--silent** – Disables all user interactive questions; this option is mainly used when writing a batch file.

Running Commands in Batch Mode

In batch mode, the DOE-C Command Line Utility takes commands from an XML file instead of requiring you to enter them interactively through the console.

Creating an XML File to Run Commands in Batch Mode

To run commands in batch mode, you must enter them into an XML file with special tagging.

1. Create a file with the XML extension and open the file in a text editor.
2. Enter the following for the first two lines:

   ```
   <?xml version="1.0" encoding="UTF-8"?>
   <commands>
   ```
3. Enter the following lines for the **login** command:

   ```
       <command name="login" sequence="1">
           <option name="url" arg="DOECSocketListenerUrl" />
           <option name="pw" arg="DOECUserPassword" />
   ```

```
        <option name="u" arg="DOECUuser" />
    </command>
```

4. For each command that you want the file to execute, enter the information for the command in an XML structure like this:

```
<command name="commandName" sequence="sequenceInteger">
    <option name="optionName1" arg="optionValue1" />
    <option name="optionName2" arg="optionValue2" />
    <option name="optionName3" arg="optionValue3" />
        ...
    <option name="optionNameN" arg="optionValueN" />
</command>
```

The sequence parameter controls the order in which the commands are executed.

5. After the last <command> entry, add the following line to terminate the file:

```
</commands>
```

Running the Command Line Utility in Batch Mode

You use a different command to run the Command Line Utility in batch mode.

Prerequisites
Create an XML file that contains the commands that you want to execute, with proper XML tagging.

1. At a command prompt, navigate to `%DOE-C_CLU_HOME%/bin/`.

2. Enter the following command:

```
execute-commandXMLFile xmlFileName
```

Where *xmlFileName* is either the full path to the file containing the commands to be executed, or the relative path to that file from the Command Line Console.

Using the Silent Option

Most of the commands in the DOE-C Command Line Utility support the silent option, which suppresses all user prompts. Generally, for batch execution, you want to do this.

When you use the silent option (-sl|--silent), make sure that you that suppressing user prompts does not have undesirable results.

Here are some examples, showing the interactive command for simplicity, that illustrate results that might be undesirable:

- `getPackages -o pac.xml -sl` - If `pac.xml` already exists, this command overwrites it without asking you for confirmation.
- `setPackageLogLevel -l DEBUG -i pac.xml` - If `pac.xml` contains more than one package, this command sets the log level for all packages to DEBUG without asking you to select the packages to be affected.

Command Summary

A concise summary of DOE-C Command Line Utility commands.

The tables below provide a concise summary of DOE-C Command Line Utility commands. For more detailed information on an individual command, refer to the reference topic on that command.

Table 5. Administrative Commands

Operation	Command
Start Command Line Utility console to enter commands interactively	`%DOE-C_CLU_HOME%/bin/clu.bat` (from a command prompt)
Run Command Line Utility to take commands from an XML file	`%DOE-C_CLU_HOME%/bin/execute-commandXMLFile` *`xmlFileName`* (from a command prompt)
Log in	`login [-h|--help] [-u|--DOEServerlUser` *`SCCUuser`*`] [-pw|--password` *`SCCUserPassword`*`] [-sl|--silent]`
Exit	`exit [-h|--help]`
Get help	`help [-h|--help]	[-a|--all] |` *`commandName`*

Table 6. Package Management Commands

Operation	Command
Deploy a package	`deploy [-h|--help] [-dir|--deployFilesDirectory` *`deploymentDirectory`*`] [-u|--technicalUser` *`SAPUserAccount`*`] [-pw|--password` *`SAPUserPassword`*`] [-sl|--silent]`
Get a list of deployed packages	`getPackages [-h|--help] [-i|in` *`inputXmlFile`*`] [-o|out` *`outputXmlFile`*`] [-sl|--silent]`
Get details for specific deployed packages	`getPackages [-h|--help] [-i|in` *`inputXmlFile`*`] [-o|out` *`outputXmlFile`*`] [-ps|--packageNames` *`nameAndVersionList`*`] [-sl|--silent]`

Set endpoint properties for a deployed package	`setEndpointProperties [-h|--help] [-i|in in-putXmlFile] [-o|out outputXmlFile] [-p|--packageName name] [-ds|--doePacketDropSize byteSize] [-u|--technicalUser SAPUserAc-count] [-pw|--password SAPUserPassword] [-t|--httpTimeout seconds] [-sl|--silent]`
Set the log level for deployed packages	`setPackageLogLevel [-h|--help] [-i|in in-putXmlFile] [-l|logLevel level] [-ps|--pack-ageNames name] [-sl|--silent]`
Get the log level for deployed packages	`getPackageLogLevel [-h|--help] [-i|in in-putXmlFile] [-o|out outputXmlFile] [-ps|--packageNames name] [-sl|--silent]`
Remove deployed packages	`removePackages [-h|--help] [-i|in inputXml-File] [-ps|--packageNames name] [-sl|--si-lent]`

Table 7. Subscription Management Commands

Operation	Command
Get information on subscriptions to de-ployed packages	`getSubscriptions [-h|--help] [-i|in in-putXmlFile] [-o|out outputXmlFile] [-ps|--packageNames name] [-sl|--silent]`
Run Command Line Utility to take com-mands from an XML file	`%DOE-C_CLU_HOME%/bin/execute-commandXMLFile xmlFileName`
Set the log level for subscriptions to a de-ployed package	`setSubscriptionsLogLevel [-h|--help] [-i|in inputXmlFile] [-p|--packageName name] [-s|--subscriptionID ID] [-sl|--silent]`
Get the log level for subscriptions to a de-ployed package	`getSubscriptionsLogLevel [-h|--help] [-i|in inputXmlFile] [-p|--packageName name] [-s|--subscriptionID ID] [-sl|--silent]`
Recover subscriptions to a deployed package	`recoverSubscriptions [-h|--help] [-i|in in-putXmlFile] [-p|--packageName name] [-s|--subscriptionID ID] [-sl|--silent]`

Resynchronize subscriptions to a deployed package	`resyncSubscriptions [-h|--help] [-i|in-putXmlFile] [-p|--packageName name] [-s|--subscriptionID ID] [-sl|--silent]`
End subscriptions to a deployed package	`endSubscriptions [-h|--help] [-i|in-putXmlFile] [-p|--packageName name] [-s|--subscriptionID ID] [-sl|--silent]`

APPENDIX B **Mobile Sales Configuration File for Windows Mobile**

System Administrators can modify the Mobile Sales system configuration for Windows Mobile in the SybaseCRM.Configuration.xml file. You must update and provision this file to the device before logging into the application (subscribing) for the first time.

The Mobile Sales installation for Windows Mobile contains two versions of this configuration file.

File name	Location on Windows Mobile device	Purpose
`SybaseCRM.Con-figuration_tem-plate.xml`	`\Program Files\Mo-bile Sales\Tem-plates`	To change system configuration settings for your company, modify this version of the configuration file. Modify options that begin with `SYS_`. Device users can modify the other options in this file by changing application settings.
`SybaseCRM.Con-figuration.xml` **Note:** Do not modify. This file is the runtime version of the configuration file.	`\Program Files\Mo-bile Sales\Data`	During application startup, the Mobile Sales application copies the version from the Templates directory to the Data directory. When device users change Settings from within the application, the new values are written to this configuration file.

Once you modify the template version of this file, you can provision it to one or more devices. For personal/developer installations, you can overwrite the default file in the Templates directory, for example, using ActiveSync. For enterprise installations, use Sybase Unwired Platform to provision the configuration file to the device. In *Sybase Unwired Platform System Administration*, see *Systems Administration > Device User Management > Device Provisioning > Afaria Provisioning and Device Management.*

Table 8. Mobile Sales for Windows Mobile system configuration options

Configuration option	Description
SYS_PushToNativeContact	Determines how native contacts synchronize with Mobile Sales contacts. • Automatic – all new Mobile Sales contacts become personal contacts; however, you must manually indicate which new personal contacts become Mobile Sales contacts. Once connected, if you change contact data in one place, it automatically synchronizes it. • Manual (default) – you must indicate which Mobile Sales contacts that you want to copy to your native contacts, and which personal contacts you want to convert to Mobile Sales contacts. Once you indicate this connection, changes to contact data in one place automatically synchronizes data between both contact instances. • None – you cannot link personal and Mobile Sales contacts.
SYS_PushToNativeCalendar	Determines how native calendar entries synchronize with Mobile Sales activities. • Automatic – new Mobile Sales activities for meetings, tasks, and appointments become personal calendar entries; however, you must manually indicate which new personal calendar entries become Mobile Sales activities. Once connected, if you change data in one place, it automatically synchronizes it. • Manual (default) – you must indicate which Mobile Sales activities that you want to copy to your native calendar, and which personal calendar entries you want to convert to Mobile Sales activities. Once you indicate this connection, changes to data in one place automatically synchronizes data between both instances. • None – you cannot link personal calendar entries and Mobile Sales activities.

Configuration option	Description
SYS_MaximumLoginAttempts	Indicates the maximum number of login attempts before the user is locked out of the application.
SYS_MaximumLoginAttemptsExceeded	Indicates whether the device user has exceeded the maximum number of login attempts. If this option is set to **true**, the user has exceeded the limit. Instruct the device user to edit this file and set it back to **false** to be able to log into the application.
SYS_MinimumMemory	Indicates the minimum memory required in MB for creating the database and initially subscribing to the SAP system. If the database already exists, this parameter is ignored. The default is 2MB.
SYS_Enable_Accounts	Enables the Accounts facet in the application. Enter **false** to disable the functionality and remove the icon from the home screen.
SYS_Enable_Contacts	Enables the Contacts facet in the application. Enter **false** to disable the functionality and remove the icon from the home screen.
SYS_Enable_Activities	Enables the Actvities facet in the application. Enter **false** to disable the functionality and remove the icon from the home screen.
SYS_Enable_Leads	Enables the Leads facet in the application. Enter **false** to disable the functionality and remove the icon from the home screen.
SYS_Enable_Opportunities	Enables the Opportunities facet in the application. Enter **false** to disable the functionality and remove the icon from the home screen.
SYS_Enable_SalesDocuments	Enables the Sales Documents facet in the application. Enter **false** to disable the functionality and remove the icon from the home screen.
SYS_Enable_Analytics	Enables the Analytics facet in the application. Enter **false** to disable the functionality and remove the icon from the home screen.

See also
- *Chapter 2, Installation Workflow for Windows Mobile* on page 3

APPENDIX C **Obtaining Help and Additional Information**

Use the Sybase Getting Started CD, SyBooks™ Online or CD or download, and online help, to learn more about this product release.

- The Getting Started CD (or download) – contains release bulletins and installation guides in PDF format, and also contains other documents or updated information not included on the SyBooks CD.
- The SyBooks CD (or download) – contains product manuals. The Eclipse-based SyBooks browser allows you to access the manuals in an HTML-based format. Some documentation is provided in PDF format, which you can access through the PDF directory on the SyBooks CD. See the *SyBooks Installation Guide* on the Getting Started CD, or the *README.txt* file on the SyBooks CD for instructions about installing and starting SyBooks.
- SyBooks Online at *http://sybooks.sybase.com/* – is an online version of SyBooks that you can access using a standard Web browser. You can browse documents online, or download them as PDFs. In addition to product manuals, the Web site also has links to EBFs/ Maintenance, Technical Documents, Case Management, Solved Cases, newsgroups, and other resources.
- Online help in the product, if available.

To read or print PDF documents, you need Adobe Acrobat Reader, which is available as a free download from the *Adobe* Web site.

Note: A more recent release bulletin, with critical product or document information added after the product release, may be available from SyBooks Online.

Index

U
uninstalling 15