Siebel Mobile Connector Guide

Siebel Innovation Pack 2014
November 2014

ORACLE

Contents

Chapter 1: What's New in This Release

Chapter 2: About the Siebel Mobile Connector

About the Siebel Mobile Connector 11

Usage Scenarios 12

- Sample Sales Voice Scenario 12
- Sample Customer Service Voice Scenario 13
- Sample Wireless Sales Scenario 13

Architecture Overview 14

How to Work with the Siebel Mobile Connector 16

Chapter 3: Installing the Siebel Mobile Connector

Installation Prerequisites 19

- Required Siebel Components 19
- Optional Siebel Components 19

Configuring Server Components 20

- Optional Server Configuration Settings 21
- After Installation of Siebel Server 22

Configuring the Siebel Mobile Connector 22

- Server Configuration Files and DLL Files 22
- Configuring the smc.cfg File 23
- Configuring the smcalert.cfg File 24
- Configuring a User Agent for the Siebel Mobile Connector Applications 25

Configuring the Application Definition 26

Chapter 4: Working with the Siebel Mobile Connector

Overview of Data Transfer Mechanisms 29

Real-Time Access to Data 29

- XML Commands and SWE Methods 30
- XML User Data 31
- Authenticating Users 35
- Retrieving Data 36

Updating and Synchronizing Data 41

Getting Updates 47
GetSMCUpdate 48

Global Language Support 50

Chapter 5: Working with the Metadata Business Service

How the Metadata Business Service Works 51
Architecture Components 52
Metadata Process Flow 52

The Metadata Business Service API 53
CanInvokeMethod 54
InvokeMethod 55
CreateStyleSheets 55
GetApplicationDef 56
LoadSubscription 58
MetadataCleanup 59
SaveSubscription 60

Using the Reference Configuration Sample 62
Performing Common Tasks 62
Applying Style Sheets 64

Reference Configuration Sample User Interface 64
Welcome Screen 64
Siebel Applications Screen 65
SMC Alert Welcome Screen 66
Load Subscription Screen 67
Metadata Extractor Screen 67
Subscription Configuration Screen 69
Style Sheets Screen 71

Chapter 6: Working with the Alert Business Service

How the Alert Business Service Works 73
Alert Process Flow 74

Configuring Alerts 74
Sample Alert Configuration Scenario 75

Appendix A: SMC Application Definition Quick Reference

SMC Account Screen 79
SMC Account View 80
SMC Account Detail View 80

SMC Account Detail - Contacts View 81
SMC Account Detail - Opportunities View 81
SMC Account Detail - Activities View 81
SMC Account Detail - Service Requests View 81
Account Attachment View 82
SMC Account List Applet 82
Account Form Applet 83
SMC Account Contact List Applet 84
SMC Oppty List Applet 85
SMC Account Activity List Applet 86
SMC Account Service List Applet 87
Account Attachment Applet 88
Account Entry Applet 89

SMC Activity Screen 91
SMC Activity View 92
SMC Activity Detail View 92
SMC Activity Detail - Contacts View 92
Activity Attachment View 93
SMC Activity List Applet 93
Activity Form Applet 94
SMC Activity Contact List Applet 97
Activity Attachment Applet 97

SMC Contact Screen 98
SMC Contact View 99
SMC Contact Detail View 99
SMC Contact Detail - Opportunities View 100
Contact Attachment View 100
SMC Contact Detail - Service Request View 100
SMC Contact Detail - Accounts View 100
SMC Contact Detail - Activities View 101
SMC Contact List Applet 101
Contact Form Applet 102
SMC Oppty List Applet 104
Contact Attachment Applet 106
SMC Service Request List Applet 106
SMC Contact Account List Applet 107
SMC Activity List Applet 108

SMC Opportunity Screen 110
SMC Opportunity View 111
SMC Opportunity Detail View 111
SMC Opportunity Detail - Contacts View 111

SMC Opportunity Detail - Activities View 112
Opportunity Attachment View 112
SMC Opportunity List Applet 112
Opportunity Form Applet 113
SMC Opportunity Contact List Applet 114
SMC Opportunity Activity List Applet 115
Opportunity Attachment Applet 116
SMC Service Request Screen 117
SMC Service Request View 118
SMC Service Request Detail View 118
Service Request Detail View 118
Service Request Detail View with Attachments 118
SMC Service Request List Applet 119
Service Request Detail Applet 119
Service Request Activity List Applet 121
Service Request Attachment List Applet 122
SMC eService Request Screen 123
SMC eService Request View 124
SMC eService Request List Applet 124
SMC eService Order Screen 125
SMC eService Order View 126
SMC eService Orders List Applet 126
SMC eOrder Line Items List Applet 127
SMC Branch Locator Screen 128
SMC Branch Locator View 129
SMC Branch Locator 129
SMC Employee Screen 130
SMC Employee View 131
SMC Employee ReadOnly List Applet 131
SMC Responsibility Screen 132
SMC Responsibility View 133
SMC Responsibility List Administration Applet 133
SMC View List Administration Applet 134
SMC User List Administration Applet 134

Appendix B: Sample XML and XSL Output

Sample XML Screen Representation 137
Sample XML Alert Message 140
Sample XSL Style Sheets 142

Appendix C: Siebel Mobile Connector Troubleshooting Guide

Troubleshooting Key 147

Troubleshooting Questions 148

Appendix D: Document Type Definition

DTD for Application Definition 155

DTD for Subscriptions 159

Index

1 What's New in This Release

What's New in Siebel Mobile Connector Guide, Siebel Innovation Pack 2014

No new features have been added to this guide for this release. This guide has been updated to reflect only product name changes.

NOTE: Siebel Innovation Pack 2014 is a continuation of the Siebel 8.1/8.2 release.

2 About the Siebel Mobile Connector

In this chapter, you will find an overview of Oracle's Siebel Mobile Connector and a brief explanation of its architecture. Additionally, scenarios are given to provide understanding of the Siebel Mobile Connector from the point of view of the users of partner applications. Finally, there is information about this guide and additional documentation.

About the Siebel Mobile Connector

The Siebel Mobile Connector allows partners and customers to create voice, wireless, and other applications with Siebel Business content. The Siebel Mobile Connector is a standards-based API that delivers well-formed XML from an optimized Siebel application definition. This gives the calling application a definition of user interface and user data in XML format.

Siebel partner and customer application developers can give mobile users real-time or near real-time access to critical Siebel Business information through a variety of mobile devices. By using speech or a wireless application on a mobile phone or personal digital assistant, users can view, edit and create information in their companies' Siebel Business repository. Siebel data and data from other applications can be combined in the same user interface. For example, employees, partners and customers can:

- Update sales opportunities
- Search for account information
- Access calendar and contact details
- Review order and parts status
- Respond to service requests

The Metadata Business Service allows customers and partners to filter data offered through the pre-configured application definitions without having to use Siebel Tools to permanently change the application configuration. The Siebel Mobile Connector generates style sheets to limit the data returned from an applet. The style sheets are stored as XSL documents on the Siebel Server. The pre-built Siebel Mobile Connector application definition has been optimized for mobile applications. These optimized views are aimed at improving performance on mobile devices. Because the Siebel Mobile Connector handles the details of abstracting the lower-level data model, it may not be necessary for third-party application developers to rewrite their code following an upgrade to Siebel applications. Thus, the upgrade path for applications using the Siebel Mobile Connector becomes relatively independent of the Siebel upgrade path.

If you need to customize the application definition beyond applying style sheets, Siebel Tools can be used. The Siebel Mobile Connector uses the same development toolset (Siebel Tools), and the same logical data model as all other Siebel Business Applications.

Additionally, the Siebel Mobile Connector provides an Alert Business Service to communicate changes to specific business components. Notifications can be pushed to employees, partners or customers who spend a majority of their time outside the office. This business service creates an XML document that can be pushed to customer or partner-developed mobile applications. For example, the reassignment of a service request from one service technician to another triggers a workflow; in this process the Alert Business Service creates an XML document with the relevant, predefined data and sends it to the mobile application, thereby allowing the application to dispatch a wireless message to both parties indicating their reassignment.

Usage Scenarios

This section provides an understanding of the Siebel Mobile Connector from the point of view of the users of partner applications. These usage scenarios are here for illustrative purposes. Siebel Alliance partners who have built validated solutions can be found listed on Oracle's Web site: http://www.oracle.com.

Sample Sales Voice Scenario

The Siebel Mobile Connector can be used to create a real-time voice interface to Siebel data. In the following example, a salesperson is using a voice interface to Siebel Sales.

Joelle Zorica is a salesperson. She is currently on her way to visit John Hiatt, a very important customer; however, she has hit a traffic jam. She does not have the customer's phone number with her so she uses a Sales Voice application to access this information. The Sales Voice application uses the Siebel Mobile Connector interface to retrieve data located in a Siebel data repository at the company offices.

1. Joelle calls into the Sales Voice application from a mobile phone.
2. The system greets her with "(h)ello. Welcome to the Sales Voice application. Please say your user ID number or enter it using the keypad."
3. Joelle responds with her user ID number.
4. The system responds "(p)lease enter your PIN."
5. Joelle responds with her PIN number and the system authenticates her log in as a valid user of the Siebel Sales system.
6. The system responds "(y)ou have new leads. Would you like to go to Opportunities, Contacts, Accounts, or Calendar?"
7. Joelle: "Contacts."
8. System: "You are in Contacts. What opportunities do you want to look up?"
9. Joelle: "Look up John Hiatt."
10. System: "John Hiatt is found."
11. Joelle: "Call John Hiatt."
12. The Sales Voice system places a call to John Hiatt and logs off Joelle from the system.

Sample Customer Service Voice Scenario

In the following example, a customer is able to be served by using a voice interface to Siebel Call Center.

Allan Street's refrigerator needs repair. He has multiple channels for communicating with the refrigerator's manufacturer: telephone, Web site, or even a wireless application. In this particular case, Allan interacts with the manufacturer's Call Center and Customer Service Voice Application. The Customer Service Voice Application uses the Siebel Mobile Connector interface to retrieve data from the company's Siebel Call Center application.

1. Allan Street calls the customer support line of the refrigerator's manufacturer to place a service request to have a service technician come repair his refrigerator. This service request is entered into Siebel Call Center, prioritized, and routed to field service centers or dispatchers. Allan receives his service request ticket number so he can check the status of his request.
2. A few hours later, Allan calls into the Customer Service Voice application to check on the status of his service request.
3. The system greets him with "(h)ello. Welcome to the Customer Service Voice application. Please say your ticket number or enter it using the keypad."
4. Allan speaks his ticket number and the system authenticates his log in as an anonymous user. The Customer Service Voice application requests the information for this ticket from the Siebel Call Center application.
5. The system plays back the ticket information: "Your ticket number is 654321. Your request is regarding repair of refrigerator model RF1. Your order is currently assigned for repair tomorrow at 1 p.m."
6. Allan decides that the description does not have enough detail, so he updates the description with the various sounds his refrigerator is making (the description is attached as an audio file). The problem is worse than he thought, so Allan escalates the service request since the repair technician is not scheduled to arrive until tomorrow afternoon.
7. The Customer Service Voice application records Allan's changes and updates the ticket information in the Siebel database. Because the ticket's priority was escalated, a service manager is alerted through email to the change. Allan's changes have triggered an alert condition that is monitored by the Alert Business Service. Aware of the new information that Allan entered into the Customer Service Voice application, the service manager assigns a service technician to pay Allan Street an immediate visit.
8. Allan completes his phone call and the Customer Service Voice application logs him off automatically.

Sample Wireless Sales Scenario

In the following example, a salesperson is using a wireless interface to Siebel Sales.

Maria Smith is a salesperson working outside the office. While she is on a sales call, the regional manager assigns Maria an opportunity for a very important prospect. The Wireless Sales application uses the Siebel Mobile Connector interface to retrieve data from the company's Siebel Sales application.

1. Maria receives a Short Message Service (SMS) message on her personal digital assistant. The message tells her that a new sales opportunity has been assigned to her and is awaiting her action to accept or reject the opportunity. She is able to accept the opportunity through two-way SMS. However, she wants to get more details.
2. To get more details about the opportunity, Maria enters her login and password into an HTML form on the Web page displayed by the wireless browser running the Wireless Sales application. The system authenticates her login as a valid user of the Siebel Sales system.
3. The system presents her with a Siebel user interface optimized for display in a mobile environment.
4. Maria taps the Opportunities screen, and a screen is displayed with the data for her opportunities.
5. Maria queries for new Opportunities and finds the new lead assigned to her. She reviews the details of the new prospect and places a call to the primary contact to begin the sales cycle.
6. When she accepts the opportunity, the Wireless Sales application sends the update for the opportunity to the Siebel database.

Architecture Overview

The Siebel Mobile Connector allows application developers to create applications that query for (or pull) information, get information pushed to the application, or create, edit or update information in the Siebel database. The following components are involved in transactions using the Siebel Mobile Connector:

A Siebel database. This is the database that users of the mobile application will access.

The Siebel Server. These components execute business logic for the Siebel application and provide an XML interface between third-party applications and the Siebel database. The Siebel Server components include the Data Manager, Object Manager, Siebel Web Engine and the Siebel Mobile Connector.

The Siebel Web Engine. Siebel Web Engine is a component of the Siebel Server that makes possible the deployment of Business applications in HTML, WML and XML. A Web browser client interacts with the Siebel database through Siebel Web Engine. The Siebel Web Engine contains the XML Web Interface that processes XML requests.

Siebel Mobile Connector. The Siebel Mobile Connector contains an optimized application definition, the Alert Business Service, the Metadata Business Service, the GetSMCUpdate method within SWE and the Reference Configuration sample application. Third-party application developers can use the Siebel Mobile Connector to access sales, service, or self-service data, create style sheets to filter Siebel data, retrieve updates, and generate and send alert (push) workflows. The Siebel Mobile Connector uses the XML Web Interface of SWE to retrieve information from the Siebel Mobile Connector application definition (or any other Siebel application definition).

Siebel Business Process Administration. This is a component that allows alerts to be sent to a third-party application server. It is a business application that can be customized by defining and managing the workflows that the alerts are based on.

A Reference Configuration Sample. This sample is provided to show how to access the Metadata Business Service. The sample generates style sheets used by the third-party application and alerts based on defined business processes. It also allows the application developer to configure smcalert.cfg, the configuration file used by the Alert Business Service to determine the transport mechanism. It is hosted by an enterprise on a Microsoft Windows 2000 Web server. Third parties may wish to expose this capability within their own toolset, allowing developers to configure the third-party application.

A third-party application server. This middleware application server exposes the infrastructure necessary for building mobile applications between the Siebel application and the end user. The middleware application server is responsible for queries for and retrieval of Siebel data. It interfaces with the Siebel Web Engine XML Web Interface using XML commands and presenting the data as required to the end-user. For example, a voice application server would contain the necessary telephony, speech recognition, and text-to-speech capability to interface with a user though a speech application. Additionally, a wireless online or offline application would contain the necessary client-server queuing software to allow the storage and forwarding of messages from the server to the client or vice versa.

A mobile client. In deployments of applications in wireless environments, Siebel data is accessed by users with client software residing on a mobile device such as a personal digital assistant or mobile phone. The client software is capable of accessing the third-party application and displaying a user interface in HTML, WML or other mark-up language. It is not necessary that the application platform be a mobile device; other platforms can also be used with the Siebel Mobile Connector. In the Siebel architecture, no components are hosted on the client.

Figure 1 illustrates the architecture of a system using the Siebel Mobile Connector to provide access to the Siebel database from a third-party mobile application.

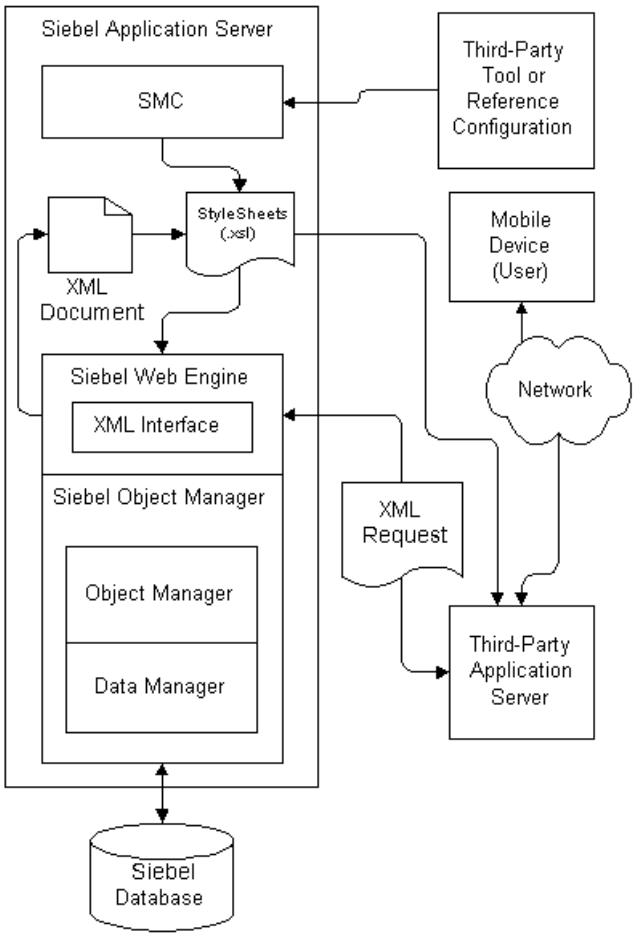


Figure 1. Siebel Mobile Connector Architecture

Each block in Figure 1 represents a separate machine, although some of the components shown separately could be installed on a single machine.

How to Work with the Siebel Mobile Connector

Third-party applications created with the Siebel Mobile Connector use standard Web protocols or specific Siebel interfaces such as a Java Data Bean or the COM Data Control to send and retrieve data between users and the Siebel database. These steps describe the flow in greater detail and assume that real-time access to Siebel data is available.

1. Using the third-party application's user interface, the user requests information residing in the Siebel database.
2. The third-party application passes the information requested by the user to Siebel Web Engine (SWE) in the form of an XML document. The request is made through HyperText Transfer Protocol (HTTP), Java Data Bean, COM Data Control or any Siebel object interface.
3. The Siebel Mobile Connector, which is part of the Siebel Server, invokes SWE to retrieve information from the Siebel database.
4. This information is then passed back through Siebel Object Manager to SWE.
5. When SWE has the requested data, it returns it in the form of an XML document to the third-party application. If less than the total data set is wanted, the request specifies a style sheet that should be applied to the data. The style sheets are located at the Siebel Server.
6. The third-party application parses the XML document and presents the Siebel data to the user in its own user interface.

NOTE: If your application provides online or offline capabilities, it must have the capability for storing and forwarding messages. In other words, your application must have a feature that queues messages between server and client, allowing the exchange of messages.

For more detailed information on how the business services of the Siebel Mobile Connector work, see "How the Metadata Business Service Works" on page 51 and "How the Alert Business Service Works" on page 73.

3 Installing the Siebel Mobile Connector

The Siebel Mobile Connector is a separately-licensed Siebel product option that is integrated into the architecture of Siebel 7.5, sharing the same Siebel Server, tool set (Siebel Tools), and installer as the rest of the Siebel Business Applications suite. As such, this document focuses on the incremental steps required to enable the Siebel Mobile Connector as part of installing Siebel Business Applications.

The Siebel Mobile Connector supports the same platforms supported by other Siebel Business applications. Operating systems, databases, and browsers supported by other Siebel Business Applications are supported by the Siebel Mobile Connector.

Installation Prerequisites

Before you begin installing the Siebel Mobile Connector, make sure that you have the appropriate license keys for the number of users that your enterprise plans to support. You must enter the license key for the Siebel Mobile Connector in order to use the product. The Siebel Mobile Connector components are installed automatically during the installation of Siebel Server and Siebel Web Engine.

For more information, see the *Siebel Server Installation Guide* for the operating system you are using.

Required Siebel Components

The Siebel Mobile Connector requires the installation of the following components:

Siebel Server 7.5. This component must include the Gateway Name Server and Siebel Web Engine.

Siebel Business Application Integration. This component is necessary for the Alert Business Service to invoke an appropriate outbound transport method.

Siebel Business Process Management. This component is necessary for the Alert Business Service and provides access to the Business Process Administration component.

Optional Siebel Components

Installation of the following components is optional:

Siebel Tools. This component is necessary for configuring the Siebel Mobile Connector views.

COM Data Control. This component must be installed on the Siebel server if your application will use this method for accessing the Siebel XML Web Interface. Also, it must be installed on the machine where you are running the Reference Configuration Sample application. This control should be automatically installed when Siebel eAI is installed. You can verify that the control is installed by viewing the registry and checking to see whether SiebelDataControl.SiebelDataControl.1 is registered.

Java Data Bean Interface. This component must be installed on the Siebel Server if your application will use this method for accessing the Siebel XML Web Interface. This control should be automatically installed when Siebel eAI is installed. For more information on the Java Data Bean Interface, see these Siebel JAR files included with your installation: SiebelJI_Common.jar and SiebelJI_*lang*.jar (where *lang* is the Language Pack you want to deploy) located in \sea75x\siebsrvr\CLASSES.

Configuring Server Components

During the installation process for the Siebel Server, you must enable the Siebel Sales component group. As shown in Figure 2, you select this option in the Enable Component Groups dialog box that appears during the Siebel Server configuration process. If you do not enable this component group, you will not be able to use the Siebel Mobile Connector.

NOTE: If you have already installed the Siebel Server, you may enable these components through Siebel Sales or any other Siebel Business application with Server Administration. For information, see "After Installation of Siebel Server" on page 22.

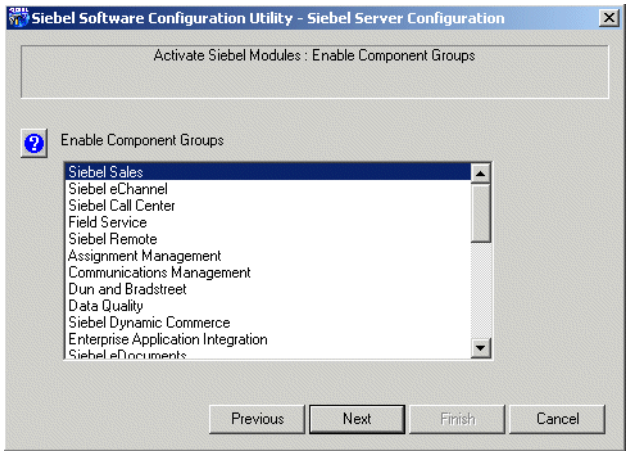


Figure 2. Siebel Sales Component Group

Optional Server Configuration Settings

During the installation process for Siebel Server, you can also enable the Workflow Management component group and the Business Application Integration component to enable the Alert Business Service. As shown in Figure 3 and Figure 4 on page 21, you select these options in the Enable Component Groups dialog box that appears during the Siebel Server configuration process. If you do not enable these component groups, you will not be able to send and receive alerts.

NOTE: If you have already installed Siebel Server, you may enable this component through Siebel Sales or any other Siebel Business application with Server Administration. For information, see "After Installation of Siebel Server" on page 22.

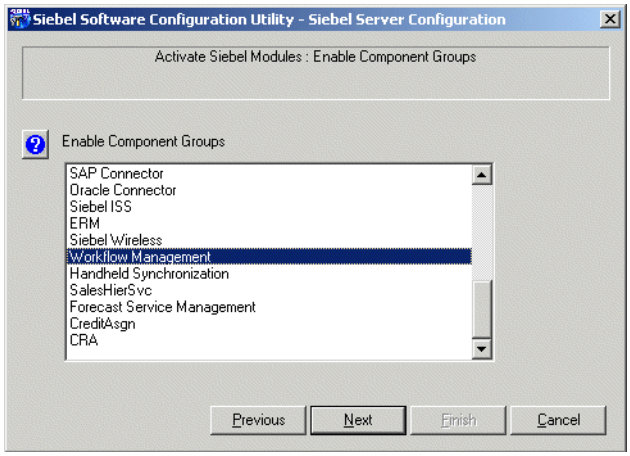


Figure 3. Workflow Management Component Group

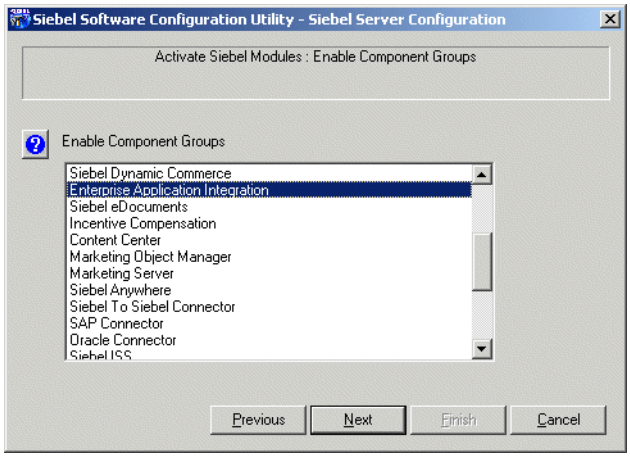


Figure 4. Enterprise Application Integration Component Group

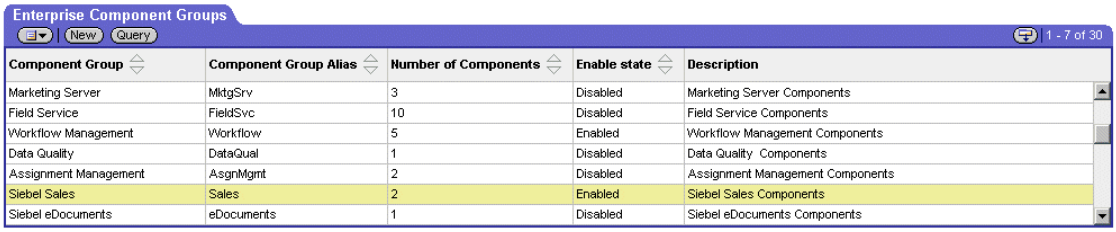
After Installation of Siebel Server

If you have already installed Siebel Server, you may enable the Workflow Management component and Siebel Sales component group through Siebel Sales or any other Siebel Business application with Server Administration.

To configure the Siebel Server settings

1. From the application-level menu, choose View > Site Map > Server Administration > Enterprise Configuration.
2. Query for Component Group = Siebel Sales.
3. From the drop-down list, select Enable Component Group.

   The value of the Enable State column is Enabled.



Component Group	Component Group Alias	Number of Components	Enable state	Description
Marketing Server	MktgSrv	3	Disabled	Marketing Server Components
Field Service	FieldSvc	10	Disabled	Field Service Components
Workflow Management	Workflow	5	Enabled	Workflow Management Components
Data Quality	DataQual	1	Disabled	Data Quality Components
Assignment Management	AsgnMgmt	2	Disabled	Assignment Management Components
Siebel Sales	Sales	2	Enabled	Siebel Sales Components
Siebel eDocuments	eDocuments	1	Disabled	Siebel eDocuments Components

4. Enable the following Server Components: Enterprise Application Integration and Workflow Management.

   NOTE: This step is optional. It is only necessary if you want to enable the Alert Business Service.

5. Select the Batch Comp Admin tab and click Synchronize.

   NOTE: Synchronization may take several minutes to complete.

6. Restart Siebel Server.

Configuring the Siebel Mobile Connector

This section describes how to modify the server configuration files to enable the Siebel Mobile Connector to work in your environment.

Server Configuration Files and DLL Files

Various server configuration files and DLL files are created during the installation of the Siebel Mobile Connector. These files are located in the *drive*:*dir_name*\siebsrvr\bin\enu directory (where *drive* is the drive, and *dir_name* is the directory where Siebel Server was installed):

smc.cfg for Mobile Connector

smcalert.cfg for Alert Business Service

Two DLL files are created for Mobile Connector during installation. These files are located in the *drive*:*dir_name*\siebsrvr\bin directory (where *drive* is the drive and *dir_name* is the directory where Siebel Server was installed):

sscalt.dll (sscalt.so on UNIX)

ssmdbldr.dll (ssmdbldr.so on UNIX)

NOTE: On UNIX systems, file locations will be different. In a typical install, the smc.cfg and smcalert.cfg files are located in a directory such as /sea75x/siebsrvr/bin/enu. The files sscalt.so and ssmdbldr.so are located in a directory such as /sea75x/siebsrvr/lib.

Configuring the smc.cfg File

The smc.cfg file contains parameters that may be configured before using the Siebel Mobile Connector application definition. However, it is not recommended to change any parameters. Many of the parameter values contained in this file are read from the Gateway Server configuration files and do not need modification.

To set the interactivity mode

1. Go to *drive*:*install_dir*\siebsrvr\BIN\ENU.

   Where:

   drive = the drive where Siebel Server is installed.

   install_dir = the directory where you installed Siebel Server.
2. Open the smc.cfg file in a text editor such as Notepad.
3. Locate the section with the [SWE] parameters.
4. Set HighInteractivity to FALSE for standard interactivity mode or TRUE for high interactivity mode.
5. Save the file.

   If you are finished with configuration, then restart the Siebel Server.

You can set the number of list rows returned by a query to a value other than the default (seven records). While the default is acceptable for a typical mobile application, your application could have special needs where it would be desirable to set a different value for the default. For example, if you are enabling a wireless browser application on a phone, you may only want four records to display on the small screen size. Or, if your application will be extracting dynamic grammars, you may want to retrieve a larger number of records when doing this batch process (for instance, 100 records per query).

To set the number of list rows returned by a query

1. Go to *drive*:*install_dir*\siebsrvr\BIN\ENU.

   Where:

   drive = the drive where Siebel Server is installed.

   install_dir = the directory where you installed Siebel Server.

2. Open the smc.cfg file in a text editor such as Notepad.
3. Locate the section with the `NumberOfListRows` parameters.

   By default, the value is 7. Change the value of this parameter if you want to specify a different number of rows.

4. Save the file.

   If you are finished with configuration, then restart the Siebel Server.

NOTE: For information about changing parameter values in the smc.cfg file that are defined during the configuration of the Siebel Server or Gateway Server, see *Siebel Server Administration Guide*.

Configuring the smcalert.cfg File

The smcalert.cfg file should be configured before using the Alert Business Service. This file specifies the transport mechanism to be used by the Alert Business Service. If a transport mechanism is specified here, it is used by default for all alerts created by the Siebel Mobile Connector. However, it is possible to specify a transport mechanism in the workflow for an alert by overriding the settings in the smcalert.cfg file. For instructions, see the "Configuring Alerts" on page 74.

The transport mechanisms allow the transportation of messages between another system the Siebel Business Application Integration (eAI) environment. Alert Business Service supports all the transport mechanisms available within eAI, including MQSeries, MSMQ, HTTP, Java Data Beans, SAP IDOC, SAP BAPI and others.

An example of a file configuration is as follows:

```
[EAI MSMQ Transport]

MsmqPhysicalQueueName=fromsiebel

MsmqQueueMachineName=machine1701


[SMC Alert]

WorkflowDelete = SMCAlert-Delete.xml

WorkflowInsert = SMCAlert-Insert.xml

WorkflowOlfValue = SMCAlert-OldFieldValue.xml

WorkflowNewValue = SMCAlert-NewFieldValue.xml
```

To set the default transport mechanisms

1. Go to *drive*:*install_dir*\siebsrvr\BIN\ENU.

   Where:

   drive = the drive where Siebel Server is installed.

   install_dir = the directory where you installed Siebel Server.

2. Open the smcalert.cfg file in a text editor such as Notepad.

3. Add a parameter to the file for each transport mechanism you want to use for the Alert Business Service.

   For each added transport mechanism, it is also necessary to add the required parameters used for configuring it.

   The MSMQ transport mechanism is displayed by default.

   You can configure MSMQ for use with the Alert Business Service by entering the name of the MSMQ Queue for *MsmqPhysicalQueueName*, and entering the machine that owns the queue specified by the physical queue name for *MsmqQueueMachineName*. You can also set any optional parameters that you want to configure.

4. Save the file.

   If you are finished with configuration, then restart the Siebel Server.

NOTE: You can also use the Reference Configuration Sample to configure the smcalert.cfg file. For more information, see "SMC Alert Welcome Screen" on page 66.

For information on supported transport mechanisms and the parameters for each, see the eAI documentation on the *Siebel Bookshelf*, especially *Transports and Interfaces: Siebel Enterprise Application Integration*.

Configuring a User Agent for the Siebel Mobile Connector Applications

The XML Web Interface requires that a user agent be configured to identify the application. A user agent could be a Web browser or a third-party application. The user agent is denoted in the HTTP header information in XML documents sent to the XML Web Interface when third-party applications send requests to SWE.

Using the Web Browser Administration screen, you may set up the user agent and its capabilities within the Siebel application. This screen is not available within the Siebel Mobile Connector application, but is available within a core Siebel Business application such as Siebel Sales. Web browser capabilities identify what an end user's browser or application can and cannot do within the Siebel Web Engine.

The Siebel Mobile Connector requires that you set the capability Voice Application or Mobile Application of the browser you want to use to TRUE. This is necessary to call the GetSMCUpdate method.

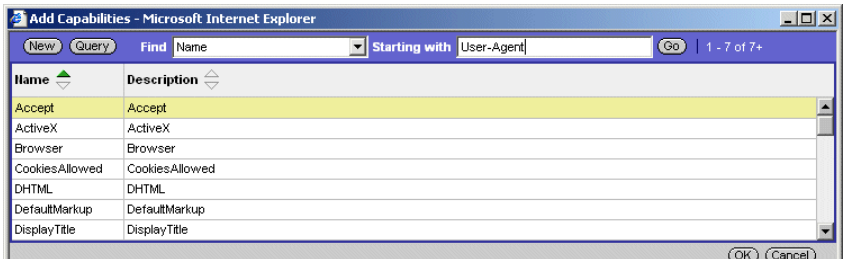
To set the VoiceApplication capability to TRUE

1. From the application-level menu, choose View > Site Map > Web Browser Administration > Browsers.
2. On the Browsers screen, query for or add the browser to be used by the application.

   For example, query for the Web browser version that your client application will be running (for example, enter IE 5.5) and select the Web browser version. If you want to add a browser instead of querying for a listed browser, click New and enter the browser's name and description.
3. Click the Capabilities tab.
4. In the Capabilities form, click New record.
5. In the Capability Name search field, enter VoiceApplication and click Go.
6. Upon retrieving VoiceApplication, click OK.

   NOTE: Although this may seem to indicate only voice applications may use this capability, this is not the case. The system does not check to see what type of application is actually connecting, only that its browser type and this capability have been registered.
7. Click Edit to edit the Capability Value field, and then enter TRUE.
8. Save the record.

   By stepping off the record, an implicit save will occur, or select Save from the drop-down list to explicitly save the record.
9. Click New to add the User-Agent Capability (refer to the preceding steps).



10. Click Edit to add the User-Agent capability value, and then save the record.

    The value that you input here must be used in the header of the XML query that will be passed to the Siebel Web Engine.
11. Restart the Siebel Server.

Configuring the Application Definition

Siebel application definitions are comprised of screens, which include various views. Views include list views, detailed views, parent views and child views. In the Siebel Mobile Connector, there are ten preconfigured screens that provide the most frequently and commonly used functions for mobile applications. Based on user responsibility type, a user can see limited screens or all of the screens when logged in through user authentication.

For a reference to the application definition, see Appendix A, "SMC Application Definition Quick Reference."

Customizing the Application Definition

The Metadata Business Service and Reference Configuration Sample give you a way to alter the application definition through XSL style sheets without permanently changing the application definition. If you want to customize the application definition, you can use Siebel Tools. You can modify the code of underlying object definitions to change the look and feel of an application.

For more information, see *Siebel Tools Reference*.

Setting User Responsibilities

As with all Siebel applications, access to views is determined by a user's responsibilities. The Siebel Mobile Connector offers six user responsibility types. Administrators must assign user responsibilities from these user types.

User Type	Responsibilities
Anonymous User - SMC	Access to: SMC Branch Locator View SMC eService Order View SMC eService Request View
Call Center Representative - SMC	Access to all views in: SMC Account Screen SMC Activity Screen SMC Contact Screen SMC Employee Screen SMC Opportunity Screen SMC Service Request Screen
Field Service Representative - SMC	Access to all views in: SMC Account Screen SMC Activity Screen SMC Contact Screen SMC Employee Screen SMC Service Request Screen

User Type	Responsibilities
Registered Customer - SMC	Access to: SMC Branch Locator View SMC eService Order View SMC eService Request View
SMC Administrator	Access to: SMC Responsibility View
Sales Representative - SMC	Access to all the views in: SMC Account Screen SMC Activity Screen SMC Contact Screen SMC Opportunity Screen SMC Employee Screen

4 Working with the Siebel Mobile Connector

This chapter provides an overview of data transfer mechanisms and information about working specifically with real-time access to data and data updates. Topics include a review of the format of XML user data and commonly used XML commands, updating and synchronizing data, and adding support for multiple languages.

Overview of Data Transfer Mechanisms

The Siebel Mobile Connector provides three mechanisms for getting data to and retrieving data from partner applications and the Siebel Server: real-time access, data updates, and pushed alerts.

Real-time access to data. You can add, modify and delete records in real-time using XML commands. User data is returned as XML documents. This capability is useful for accessing contact details, updating opportunities, accessing contacts, and so forth.

Data updates. You can query the Siebel database for new information entered for a specific user. This capability is useful for retrieving information that has changed or been added since the last time a user has logged into the application, such as new activities or new opportunities.

Pushed alerts. You can also obtain pushed alerts from the Alert Business Service. The Alert Business Service creates an XML document with the relevant, predefined data and sends it to the third-party application. For example, two service technicians could be notified that a service request has been reassigned from one to the other.

If you are using the COM Data Control or Java Data Bean to send HTTP or XML requests to SWE, then the XML output received from SWE will contain an encoding property of UTF-16. In other words, the XML header will look like this:

```
<?xml version="1.0" encoding="UTF-16"?>.
```

If the XML output must be loaded in a Web browser or an XML parser, then you must remove this property or set it to UTF-8. The header will like this:

```
<?xml version="1.0" ?> or <?xml version="1.0" encoding="UTF-8"?>.
```

For more information on UTF encoding, see My Oracle Support.

Real-Time Access to Data

This section provides an overview of working with the Siebel Mobile Connector to obtain real-time access to Siebel data from third-party applications. Mobile applications make requests to the Siebel Mobile Connector through HyperText Transfer Protocol (HTTP), Siebel COM Data Control, Siebel Java Data Bean interface or any other Siebel object interface that supports the GetService method.

For information on object interfaces including Siebel COM Data Control and the Siebel Java Data Bean interface, see *Siebel Tools Online Help*.

NOTE: Examples in this document show commands in the HTTP format. In the examples, spaces in the HTTP requests are shown replaced by a plus sign (+). It is recommended that you replace the spaces in HTTP requests with plus sign (+) symbols.

XML Commands and SWE Methods

You can add, modify and delete records in real time using XML commands. The following are some common XML commands that can be issued from third-party applications to the Siebel Mobile Connector, which in turn uses the XML Web Interface provided by SWE.

- CanInvokeMethod
- ExecuteLogin
- ExecuteNamedQuery
- GotoPage
- GotoPageTab
- GotoView
- InvokeMethod
- LoadService
- Login
- Logoff
- ReloadCT

The following are some methods commonly used with SWE:

- CollapseTreeItem
- CopyRecord
- CreateRecord
- DeleteQuery
- DeleteRecord
- Drilldown
- EditRecord
- ExecuteQuery
- ExpandTreeItem
- GetSMCUpdate
- GotoFirstSet
- GotoLastSet
- GotoNextSet
- GotoPreviousSet
- GotoView
- MoveDown
- MoveUp
- NewQuery
- NewRecord
- NextTreeItem
- PickNone
- PickRecord
- PositionOnRow
- PreviousTreeItem
- RefineQuery
- SaveQueryAs
- SelectTreeItem
- SortAscending
- SortDescending
- ToggleTo
- UndoRecord
- WriteRecord

NOTE: SWE expects the correct spelling and valid values for all commands, methods and arguments. Invalid SWE commands, methods and arguments are ignored. No error message is returned by SWE for such errors and the calling application may experience unexpected results.

For information on the XML interface to SWE, see *Siebel Tools Online Help*.

XML User Data

This section gives a summary of the most common XML tags returned to the third-party application following a request for user data to SWE. The user data is returned as an XML document.

In the Siebel database, data is contained in applets, which are contained within views, and views in turn are contained in screens. Make an XML request to the SWE to access data in the Siebel database. If you want to retrieve only data, set the parameter SWEDataOnly to TRUE. By using this flag, you can make sure the XML document contains only data tags and does not contain any user interface navigation elements such as drop-down menus, page tabs, and so on.

XML Page Content

In order to support different implementations, the Siebel Mobile Connector defines user interfaces and user data in the XML format.

In response to the XML requests of third-party applications, the Siebel Mobile Connector returns XML pages as output. The XML page output is based on the application definitions, including application, screen, view, applet and controls, that are defined in a repository. The output can be tailored to the application by including only data specified during configuration by applying XSL style sheets generated by the Siebel Mobile Connector for this purpose.

Section	Required or Optional	Description
XML version and encoding	Required	Describes the version of XML supported and the type of encoding used. Appended in all XML pages.
Application	Required	Describes the application name, such as the Siebel Mobile Connector or the Siebel Sales Enterprise, that the third-party application is connected to and interacting with. Appended in all XML pages.
User agent markup	Required	Describes the default markup language that is supported. It is based on the user-agent in the HTTP request header.
Navigation elements	Optional	Contains the following user interface information (the UI is defined in Siebel Tools): Menu, Tool bar, Screen bar, Thread bar, and Page item. The information under this tag can be turned off by specifying SWEDataOnly=TRUE in the HTTP request.

Section	Required or Optional	Description
Form definitions	Optional	Contains the user interface for predefined queries (the UI is defined in Siebel Tools). Like the navigation elements, this information is generated by default. The information under this tag can be turned off by specifying SWEDataOnly=TRUE in the HTTP request.
Active Screen, View and Applets definition and User Data	Optional	Contains the current active screen and view information, applets, and the record (user data) defined in that view. Some UI elements are contained in this section that will not be included in data only mode. This section is generated by default. Specify SWEDataOnly=TRUE in the HTTP request to return only this information.

Common XML Tags

Table 1 provides a list of common XML tags returned in user XML documents and their attributes.

Table 1. Common XML Tags and Attributes

XML Element	Description
APPLICATION	Specifies the name of the application. For example, <APPLICATION NAME="Siebel Mobile Connector"> where APPLICATION is the tag, NAME is its attribute and the Siebel Mobile Connector is the value of NAME attribute.
USER_AGENT	Specifies information about the user agent or the browser type that made the XML or HTML request.
SCREEN	Specifies information about the name and title for the currently active screen. Contained inside the APPLICATION element.
VIEW	Describes the name and title of the currently active view (similar to the SCREEN element).
APPLET	Returns additional information in addition to the name and title of the applet. Contained inside the VIEW element.
MODE	Describes what mode the applet is in. Possible values include EDIT or BASE. EDIT specifies that the applet allows modification, deletion, creation and querying of records. BASE specifies that the applet is read only and cannot be modified.

Table 1. Common XML Tags and Attributes

XML Element	Description
NO_INSERT, NO_MERGE, NO_DELETE, NO_UPDATE, NO_EXEC_QUERY	Provides a filter to what specific edit mode operations are possible for the applet. If any of these attributes are TRUE, then that particular operation is not possible. For example, if NO_INSERT attribute is TRUE then new records cannot be inserted into the applet. The third-party application can customize the associated commands based on these attributes. For example, if the NO_EXEC_QUERY attribute is set to FALSE for an applet, this indicates that the third-party application should be able to query for a contact using that applet.
CLASS	Specifies the C++ class the applet belongs to. For example, in the first sample in this section, the CLASS attribute has a value of CSSFrameListBase, which means it is a List applet. The second sample has a CLASS value of CSSFrameBase, which means it is a Form applet.
ROW_COUNTER	Gives an indication of the number of data records returned. A plus sign (+) at the end indicates that there are more records than that returned.
RS_HEADER	Contains the COLUMN element.
COLUMN	Specifies the column details for the data records.
NAME, DISPLAY_NAME, TEXT_LENGTH	Specifies information about the name, title and text length of the columns respectively.
DATATYPE	Describes what kind of data type the column represents. For example, the phone number has a data type of phone and an email has a data type of email. This information could be used by the third-party application to make a call or send an email.
REQUIRED	Specifies whether or not the column is required. This information is useful when creating new records. The third-party application can determine what field information is mandatory by looking at this attribute.
FORMAT	Specifies the format of the data. For the Date data type this attribute should contain the acceptable Date Format (refer to the following sample). For revenue and other price related fields this attribute will have the format for the dollar amount. The third-party application can use this to get or display the right information back to the user.
CALCULATED	Specifies that the column has been calculated, for example, by using mathematical expressions. The column has not been directly derived from the database tables. This information could be useful during record creation.
FIELD	Specifies the name of the FIELD element in the business component that the column refers to. The FIELD element contains the actual data. The third-party application would make use of both FIELD and COLUMN elements to get more information on the data. FIELD is useful in determining what fields to query on while fetching a particular record.

Table 1. Common XML Tags and Attributes

XML Element	Description
READ_ONLY, LIST_EDITABLE	Specifies whether the column is editable or just read only. This information could be useful to the third-party application when modifying certain columns.
NUMBER_BASE, TEXT_BASED	Indicates whether the column or field is a number or text.
RS_DATA	Contains the XML tags that hold the actual data.
ROW	Identifies the row id of the data in the attribute ROW_ID. This information is very useful in querying for a particular row of data and getting the detailed information for that row.
SELECTED	Indicates that the particular row is selected on the user interface.

Retrieving Data Only

If SWEDataOnly is set to TRUE, all elements contained within both the NAVIGATION_ELEMENTS tag and the FORM tag will not be returned. For example, this code fragment represents an XML document where SWEDataOnly is set to FALSE:

```
<?xml version="1.0" encoding="UTF-8" ?>
- <APPLICATION NAME="Siebel Mobile Connector">
    <USER_AGENT MARKUP="HTML" />
+ <NAVIGATION_ELEMENTS>
+ <FORM ACTION="/smc/start.swe" METHOD="POST" NAME="SWEForm4">
+ <SCREEN CAPTION="Accounts" ACTIVE="TRUE" NAME="SMC Account Screen">
    </APPLICATION>
```

In contrast, this code fragment represents an XML document where SWEDataOnly is set to TRUE:

```
<?xml version="1.0" encoding="UTF-8" ?>
- <APPLICATION NAME="Siebel Mobile Connector">
    <USER_AGENT MARKUP="HTML" />
+ <SCREEN CAPTION="Accounts" ACTIVE="TRUE" NAME="SMC Account Screen">
    </APPLICATION>
```

Retrieving Detailed Information About the Data

The RS_HEADER section holds detailed information about each data column. For instance, if the third-party application wants to detect if a particular column holds a phone number, then it should lookup the DATATYPE attribute in the COLUMN element (under the RS_HEADER section) and then get the data from the FIELD element. The FIELD attribute of the COLUMN element gives a link to the FIELD element, which holds the actual data.

NOTE: Using the field attributes in the RS_DATA section for data type detection is not recommended, because this information is not guaranteed to be a constant. The RS_DATA might change if the object definition, field names in this case, are changed in Siebel Tools.

Authenticating Users

Authentication is the process of verifying the identity of a user. The Siebel Mobile Connector uses the Siebel security adapter authentication architecture for authenticating Siebel application users to external partner applications. This architecture supports authentication to Microsoft Active Directory Server and LDAP-compliant directories. Additionally, partners may also create their own authentication mechanism by writing to the Siebel Security Adapter API. For more information, search for Security Adapter SDK on My Oracle Support.

Partner applications must log in to the Siebel Web Engine to instantiate a user session and must log out to terminate the session. A user's session is managed in SWE by using cookies or an authentication mode without cookies.

Logging In

Logging in to SWE is required to instantiate a new user session. The command ExecuteLogin is used to supply the user credentials and log in.

The following is an example of how to construct a log in command by using an HTTP request.

To log in to SWE

Send the ExecuteLogin command in a HTTP request to SWE, with a valid user name and password.

In the following example, your user name and password is WLEE.

```
http://localhost/smc_enu/
start.swe?SWECmd=ExecuteLogin&SWEUserName=WLEE&SWEPassword=WLEE&SWESetMarkup=XML
```

NOTE: SWESetMarkup is a required parameter. Generally, when using the Siebel Mobile Connector you can set the SWESetMarkup parameter to HTML, WML, or XML. However, you may not use the GetSMCUpdate method or receive alerts in any other markup than XML.

Logging Off

Logging off of SWE is required to terminate a new user session. The command Logoff is used to log off.

To log off from SWE

Send the Logoff command in a HTTP request to SWE.

```
http://localhost/smc_enu/start.swe?SWECmd=Logoff&SWESetMarkup=XML
```

NOTE: There is a default time-out set by the SWE engine if no user action has taken place.

Session Management

When a user logs in to the Siebel Web Engine, SWE dynamically generates session cookies or uses an authentication mode without cookies. Cookies are generated by default and include a Session ID that is used to track the session. If cookies are disabled or if a user's browser does not support cookies, then the Session ID for each page is included in its URL.

For more information, refer to *Siebel Security Guide*.

Retrieving Data

This section describes how to retrieve data from SWE by using the XML Web Interface. Included is an overview of navigating to a screen, navigating within a screen, as well as information about queries and updates.

Navigating to a Screen

Navigating to a screen is required to retrieve data about the screen's views and applets. The command GotoPageTab is used to go to a specific screen.

To navigate to a screen

1. Log in to SWE. For more information, see "Logging In" on page 35.
2. Navigate to the screen to which you want to go.

   In the following example you are navigating to the SMC Opportunity Screen:

   ```
   http://localhost/smc_enu/
   start.swe?SWECmd=GotoPageTab&SWEScreen=SMC+Opportunity+Screen&SWEDataOnly=TRUE&SWE
   SetMarkup=XML
   ```

NOTE: GotoPageTab executes the default PDQ (predefined query) for that screen.

The following is a list of screens provided in the Siebel Mobile Connector application definition to which you can navigate.

Screen Name	Display Name
SMC Account Screen	Accounts
SMC Activity Screen	Activities
SMC Contact Screen	Contacts

Screen Name	Display Name
SMC Opportunity Screen	Opportunities
SMC Service Request Screen	Service Requests
SMC eService Request Screen	eService Requests
SMC eService Order Screen	Service Orders
SMC Branch Locator Screen	Branch Locator
SMC Employee Screen	Employees
SMC Responsibility Screen	Responsibilities

Navigating Within a Screen

Navigating within a screen is required to perform an action on data from a screen's views and applets. You can use to the GotoView command to go to a particular Siebel view, where you can access the applets available to that view. The GotoView command requires the name of the view to be passed in the SWEView parameter.

To navigate to a view or applet

1. Log in to SWE and navigate to the screen to which you want to go. For more information, see "Logging In" on page 35 and "Navigating to a Screen" on page 36.
2. Navigate to the view and applet to which you want to go.

   Example (where you are navigating to the SMC Opportunity Detail - Contacts View):

   ```
   http://localhost/smc_enu/
   start.swe?SWECmd=GotoView&SWEView=SMC+Opportunity+Detail+-
   +Contacts+View&SWENeedContext=false&SWEBID=-1&SWEKeepContext=1&SWESetMarkup=XML
   ```

For a list of the view and applet names to which you can navigate, see Appendix A, "SMC Application Definition Quick Reference."

Querying Items

To perform a query, you must navigate to the screen that allows queries and then you must send two separate requests to SWE.

The two requests are:

The NewQuery command

The ExecuteQuery command

In the ExecuteQuery command block, you must specify a parameter to identify the column (the field you want to search) and a value to indicate the search criteria.

To perform a query

1 Login to SWE and navigate to the screen, view and applet to which you want to go.

For more information, see "Logging In" on page 35, "Navigating to a Screen" on page 36 and "Navigating Within a Screen" on page 37.

2 Invoke the NewQuery method.

Example (where you want to query on a field in the SMC Opportunity View):

```
http://localhost/smc_enu/
start.swe?SWECmd=InvokeMethod&SWEApplet=SMC+Opportunity+List+Applet&SWEView=SMC+Opportunity+View&SWENeedContext=false&SWEReqRowId=0&SWEBID=-1&SWEMethod=NewQuery&SWESetMarkup=XML
```

3 Invoke the ExecuteQuery method and specify a value to indicate the search criteria.

In the following example, query for a record name called IP_Webserver:

```
http://localhost/smc_enu/
start.swe?SWECmd=InvokeMethod&SWEApplet=SMC+Opportunity+List+Applet&SWEView=SMC+Opportunity+View&SWENeedContext=false&SWEReqRowId=0&SWEBID=-1&SWEMethod=ExecuteQuery&SWESetMarkup=XML&Name=IP_Webserver
```

For a list of the view anapplet names to which you can navigate, see Appendix A, "SMC Application Definition Quick Reference."

Drilling Down on Items

You can drill down on a field by specifying the name of the applet field on which you want to drill down. The detailed information about the field is retrieved from the repository. In this way you can retrieve detailed information about specific items in applets on which you have queried.

To drill down on an item

1 Login to SWE and navigate to the screen, view and applet to which you want to go.

For more information, see "Logging In" on page 35, "Navigating to a Screen" on page 36 and "Navigating Within a Screen" on page 37.

2 Invoke the DrillDown method and pass the value of the field you want to drill down to in the SWEField argument.

In the following example, drill down on the Account field:

```
http://localhost/smc_enu/start.swe?
SWECmd=InvokeMethod&SWEApplet=SMC+Opportunity+List+Applet&SWEView=SMC+Opportunity+View&SWERowId=99-27NLD&SWENeedContext=true&SWEReqRowId=1&SWEMethod=Drilldown&SWEField=Account&SWESetMarkup=XML
```

NOTE: If you want to drill down into a specific record, then specify the SWERowId parameter of the row you want to drill down to. When the SWERowId parameter is not supplied in a drill down, then SWE returns the first record in the list.

For a list of the field names on which you can drill down, see Appendix A, "SMC Application Definition Quick Reference."

Executing Pre-Defined Queries (PDQs)

You can execute pre-defined queries from your applications. You must invoke the ExecuteNamedQuery method and pass the name of the PDQ you want to apply.

To execute a PDQ

1. Log in to SWE and navigate to the screen, view and applet to which you want to go.

   For more information, see "Logging In" on page 35, "Navigating to a Screen" on page 36 and "Navigating Within a Screen" on page 37.

2. Invoke the ExecuteNamedQuery method and pass the value of the PDQ you want to use.

   Example (to use the My Activities for the Week PDQ):

   ```
   http://localhost/smc_enu/
   start.swe?SWECmd=ExecuteNamedQuery&SWEView=SMC+Activity+View&SWENeedContext=false&
   SWEQueryName=My+Activities+for+the+Week&SWEBID=-1&SWESetMarkup=XML
   ```

Retrieving Large Data Sets

It may be necessary for an application to retrieve a very large set of data, or to obtain all the records in a set. You can retrieve large data sets by using the SWESetRowCnt parameter in your request to SWE.

Set the parameter to a large number such as 100 to obtain up to 100 records. If you set the SWESetRowCnt parameter to a large number, it will take longer to get a response back from SWE and the performance may not be acceptable to your end user if this is a real-time action taken on their behalf. You can set the SWESetRowCnt parameter to a smaller number to improve performance, but check the results to see if there are additional records.

If there are:

Additional records, you can make additional requests to SWE

More rows to bring back, invoke the GoToNextSet method

To retrieve all the records in a large set

1. Login to SWE and navigate to the screen, view and applet to which you want to go.

   For more information, see "Logging In" on page 35 and "Navigating to a Screen" on page 36.

2. Navigate within the screen to which you want to query.

   If there are more than 100 records, you can set the row count to 100 to get the first set of 100 records.

   In the following example, you are navigating to the Contact Attachment View:

```
http://localhost/smc_enu/
start.swe?SWECmd=GotoPageTab&SWEScreen=SMC+Account+Screen&SWENeedContext=false&SWE
BID=-1&SWESetMarkup=XML&SWESetRowCnt=100
```

3 Examine the XML page that is returned from SWE.

NOTE: The APPLET tag contains the ROW_COUNTER attribute that indicates whether there are additional records.

In the following example, the ROW_COUNTER indicates that there are additional records by containing a plus sign (+):

```
<APPLET MODE="Edit" ROW_COUNTER="1 of 100+" NO_INSERT="FALSE" ACTIVE="FALSE"
CLASS="CSSFrameBase" TITLE="Account" ID="1" NO_MERGE="FALSE" NO_DELETE="FALSE"
NO_UPDATE="FALSE" NO_EXEC_QUERY="FALSE" NAME="Account Form Applet">
```

4 Query again, this time invoking the GoToNextSet method to obtain the next set of records.

Example:

```
http://localhost/smc_enu/
start.swe?SWECmd=InvokeMethod&SWEApplet=SMC+Account+List+Applet&SWEView=SMC+Accoun
t+View&SWENeedContext=false&SWEReqRowId=0&SWEBID=-
1&SWEMethod=GotoNextSet&SWESetMarkup=XML
```

5 Again, examine the XML page that is returned from SWE.

NOTE: The APPLET tag contains the ROW_COUNTER attribute that indicates whether there are additional records.

In the following example the ROW_COUNTER indicates that there are additional records by containing a plus sign (+):

```
<APPLET MODE="Edit" ROW_COUNTER="101 of 200+" NO_INSERT="FALSE" ACTIVE="FALSE"
CLASS="CSSFrameBase" TITLE="Account" ID="1" NO_MERGE="FALSE" NO_DELETE="FALSE"
NO_UPDATE="FALSE" NO_EXEC_QUERY="FALSE" NAME="Account Form Applet">
```

6 To get all the records in this set, continue querying in this way until there are no additional records returned (that is, when the ROW_COUNTER attribute does not contain a “+” sign).

Parsing Dates on Records

Applications may need to parse the dates and times on records to perform operations on the data. Many fields contain date stamp information according to formats specified in the Siebel application definition. Your application must read the format specified in the FORMAT attribute to parse dates on records.

In the following examples the date format is M/D/YYYY hh:mm:ss p and M/D/YYYY hh:mm p:

```
<CONTROL HTML_TYPE="Field" CAPTION="Created" SCALE="0" DATATYPE="utcdatetime" HIDDEN="TRUE"
NUMBER_BASED="FALSE" ID="1310" TYPE="TextBox" REQUIRED="TRUE" TEXT_BASED="FALSE" FORMAT="M/D/YYYY hh:mm:ss
p" CALCULATED="FALSE" ENABLED="TRUE" MAX_LENGTH="32" NAME="Created">12/31/1979 04:00:00 PM</CONTROL>
```

```
<CONTROL HTML_TYPE="Field" CAPTION="Start" SCALE="0" DATATYPE="utcdatetime" HIDDEN="FALSE"
NUMBER_BASED="FALSE" ID="1801" TYPE="TextBox" REQUIRED="FALSE" TEXT_BASED="FALSE" FORMAT="M/D/YYYY hh:mm
p" CALCULATED="FALSE" ENABLED="TRUE" MAX_LENGTH="32" NAME="Planned">8/12/1999 03:00 PM</CONTROL>
```

NOTE: The FORMAT attribute uses the standard Siebel date format specification. For example, to indicate the month of March, a single capital M indicates that the month is represented by “3”; MM indicates “03”; MMM indicates “Mar”; and MMMM indicates “March.”

Retrieving Data from Hidden Fields

In Siebel application definitions, some form applets are not entirely visible by default. On the user interface, the user must click the toggle button to switch between views of the form applet. When retrieving data from these forms, by default SWE will only return data from the visible fields. If you want data from the hidden fields, use the ToggleLayout command.

The following example shows a SWE request for toggling the layout:

```
http://localhost/
start.swe?SWECmd=InvokeMethod&SWEApplet=Account+Entry+Applet&SWEView=Account
Attachment+View&SWERowId=99-
28B1T&SWENeedContext=true&SWEReqRowId=0&SWEMethod=ToggleLayout&SWESetMarkup=XML
```

Updating and Synchronizing Data

This section describes how to update and synchronize data by using the XML Web Interface.

Adding Records

The following procedure describes how to add records to a list. You must first navigate to a screen that allows rows to be inserted, send requests to SWE to execute a new record, and write the data to the record. The commands used are NewRecord and WriteRecord.

To add a record

1. Login to SWE and navigate to the screen to which you want to go.

   For more information, see “Logging In” on page 35 and “Navigating to a Screen” on page 36.

2. Execute a new record by using the NewRecord command.

   Example:

   ```
   http://localhost/smc_enu/
   start.swe?SWECmd=InvokeMethod&SWEApplet=SMC+Contact+List+Applet&SWEView=SMC+Contac
   t+View&SWENeedContext=true&SWEReqRowId=0&SWEMethod=NewRecord&SWESetMarkup=XML
   ```

   NOTE: For a NewRecord command, the SWERowID parameter should be empty. The XML returned from this command will contain the new RowID for the record created. You will use the new RowID value returned from SWE in the next step. For an example of XML output from SWE that contains a RowID, see the example in Step 5 on page 46.

3 Fill in the fields in the user interface, and then write the data to the record. Use the WriteRecord command.

Example:

```
http://localhost/smc_enu/
start.swe?SWECmd=InvokeMethod&SWEApplet=SMC+Contact+List+Applet&SWEView=SMC+Contac
t+View&SWERowId=99-
4CESH&SWENeedContext=true&SWEReqRowId=1&SWEMethod=WriteRecord&Last+Name=Haven&Firs
t+Name=Chris&SWESetMarkup=XML
```

Modifying Records

The following procedure describes how to modify a record. You must first navigate to a screen that allows records to be modified, perform a new query, execute the query, invoke the edit record method, and write the record.

To modify a record

1 Login to SWE and navigate to the screen, view and applet to which you want to go.

For more information, see "Logging In" on page 35, "Navigating to a Screen" on page 36 and "Navigating Within a Screen" on page 37.

2 Query for the record you want to modify.

For more information, see "Querying Items" on page 37.

CAUTION: If you do not use a primary key to perform the query, several records may be returned in the response. The record you want to modify may not be the one selected.

3 Write the record. You must invoke the WriteRecord method to modify the record.

In the following example, modify a Job Title to read QA:

```
http://localhost/smc_enu/
start.swe?SWECmd=InvokeMethod&SWEApplet=SMC+Contact+List+Applet&SWEView=SMC+Contac
t+View&SWERowId=99-
27NLD&SWENeedContext=true&SWEReqRowId=1&SWEMethod=WriteRecord&SWESetMarkup=XML&Job
+Title=QA
```

In the following example modify a record with the fields Job Title, Work Phone #, and Email Address:

```
http://localhost/smc_enu/
start.swe?SWECmd=InvokeMethod&SWEApplet=SMC+Contact+List+Applet&SWEView=SMC+Contac
t+View&SWERowId=99-
27NLD&SWENeedContext=true&SWEReqRowId=1&SWEMethod=WriteRecord&SWESetMarkup=XML&Job
+Title=QA+Engineer&Work+Phone #=4255551212&Email+Address=someone@oracle.com
```

Deleting Records

The following procedure describes how to delete a record. You must first navigate to a screen that allows records to be modified, perform a new query, execute the query, and delete the selected record.

To delete a record

1. Login to SWE and navigate to the screen, view and applet to which you want to go.

   For more information, see "Logging In" on page 35, "Navigating to a Screen" on page 36 and "Navigating Within a Screen" on page 37.

2. Query for the record you want to delete.

   For more information, see "Querying Items" on page 37.

   CAUTION: If you do not use a primary key to perform the query, several records may be returned in the response. The record you want to delete may not be the one selected.

3. Delete the selected record by using the DeleteRecord to access the record by its primary key (in this case, the RowID).

   Example:

   ```
   http://localhost/smc_enu/
   start.swe?SWECmd=InvokeMethod&SWEApplet=SMC+Contact+List+Applet&SWEView=SMC+Contac
   t+View&SWERowId=0-
   10LMD&SWENeedContext=true&SWEReqRowId=0&SWEMethod=DeleteRecord&SWEC=5
   ```

Synchronizing Records One-Way

In some instances, a third-party application may require that a local copy of a subset of Siebel data be synchronized with data residing on the third-party application server. Such one-way synchronization of records must be handled by third-party applications. It is necessary for such applications to track the date and time that a user last synchronized data on the mobile device with the application server, and the date stamp must be passed in the GetSMCUpdate method to obtain any changed records since the date stamp.

An example of how synchronization could be used is a voice application. In order to recognize unique names or words (such as accounts or contacts), the speech recognition system must compile these words into phonemes. This is usually done on a batch process based on how frequently the data is expected to change. Using the GetSMCUpdate method, you can compile a smaller subset of information for a user giving them the ability to get real-time information from the Siebel Business application.

For more information about the GetSMCUpdate method, see "GetSMCUpdate" on page 48.

NOTE: The ability to automatically synchronize data records is a feature that may be available in future releases of the Siebel Mobile Connector.

Uploading Files

Applications may require that files be uploaded to the Siebel database. For example, mobile voice applications may allow users to update descriptions or add comments by capturing speech in an audio file and attaching it to the record.

NOTE: This procedure cannot be done by sending HTTP requests in a browser. Instead, uploading files must be done programmatically, so the application that uploads the files can modify the Content-Type of the HTTP request and send the file according in the appropriate format for file uploads.

To upload a file

1 Log in to SWE and navigate to the screen, view and applet to which you want to go.

For more information, see "Logging In" on page 35, "Navigating to a Screen" on page 36 and "Navigating Within a Screen" on page 37.

The Content-Type of the HTTP request must be set to application/x-www-form-urlencoded for each step of this procedure, except where noted.

2 Create a new record by invoking the NewRecord method to make a record for the file you want to attach.

In the following example Content-Type is application/x-www-form-urlencoded:

```
http://localhost/smc_enu/
start.swe?SWECmd=InvokeMethod&SWEApplet=Account+Attachment+Applet&SWEView=Account+
Attachment+View&SWERowId=&SWENeedContext=true&SWERowIds=SWERowId0=10-
5NIG6U&SWEReqRowId=0&SWEMethod=NewRecord&SWEC=3&SWESetMarkup=XML
```

NOTE: The SWERowId parameter is not required here and can be empty (a value for this parameter will be returned). However, SWERowId0 is a required parameter. SWERowID0 is the id of the parent row and SWERowId is the child's id. The Row IDs of the parent and child can be obtained from the XML output returned from the previous request.

3 Edit the field by invoking the EditField method and modifying the record for the file you want to attach.

In the following example, Content-Type is application/x-www-form-urlencoded:

```
http://localhost/smc_enu/
start.swe?SWECmd=InvokeMethod&SWEApplet=Account+Attachment+Applet&SWEW=0&SWEView=A
ccount+Attachment+View&SWERowId=&SWEField=AccntFileName&SWEDIC=true&SWENeedContext
=true&SWERowIds=SWERowId0=10-
5NIG6U&SWEH=0&SWEReqRowId=1&SWESP=true&SWEMethod=EditField&SWEC=4&SWESetMarkup=XML
```

NOTE: You will need to supply the SWERowID parameter returned in the XML output from Step 2. SWE requests that require a RowID will not work if the SWERowId parameter is not supplied.

4 Attach the file to the record by invoking the WriteRecord method. Set the Content-Type of the HTTP request containing the attachment to multipart/form-data for this step of the procedure.

In the following example, Content-Type is multipart/form-data:

```
Content-Type: multipart/form-data; boundary=BbC04y
```

```
--BbC04y
      Content-Disposition: name="SWECmd"
InvokeMethod
--BbC04y
Content-Disposition: name="SWEApplet"
File Popup Applet
--BbC04y
Content-Disposition: name="SWEView="
Account Attachment View
--BbC04y
Content-Disposition: name="SWERowId"
10-5O167A
--BbC04y
Content-Disposition: name="SWENeedContext"
false
--BbC04y
Content-Disposition: name="SWERowIds"
SWERowId0=10-5NIG6U
--BbC04y
Content-Disposition: name="SWEReqRowId"
1
--BbC04y
Content-Disposition: name="SWEP"
19_Account+Attachment+Applet9_EditField9_10-5O167AD_AccntFileName1_4
--BbC04y
Content-Disposition: name="SWEMethod"
WriteRecord
--BbC04y
Content-Disposition: name="SWESetMarkup"
XML
```

```
--BbC04y

Content-Disposition: name="SWEC"

4

--BbC04y

Content-Disposition: name="_SweFileName"; filename="file1.txt"

   Content-Type: text/plain

... contents of file1.txt ...

--BbC04y--
```

TIP: For additional information about HTTP file uploads, the relationship between multipart/form-data and other content types, performance issues, and so on, see the documentation available at http://www.w3.org/.

5 After you upload the file, examine the XML content of the user data you get back from SWE and make a note of the SWERowId parameter.

You will need the value of this parameter in the next step.

In the following example, the RowId is 10-5O167A:

```
- <ROW ROWID="10-5O167A" SELECTED="TRUE">

<FIELD VARIABLE="AccntFileName" NAME="AccntFileName" />

   <FIELD VARIABLE="AccntFileSize" NAME="AccntFileSize" />

   <FIELD VARIABLE="AccntFileExt" NAME="AccntFileExt" />

   <FIELD VARIABLE="AccntFileDate" NAME="AccntFileDate" />

   <FIELD VARIABLE="AccntDockStatus" NAME="AccntDockStatus"><img width="11"
   alt="ENU_white.gif" src="images/white.gif" border="0" height="11"/></FIELD>

   <FIELD VARIABLE="AccntFileDockReqFlg" NAME="AccntFileDockReqFlg" />

   <FIELD VARIABLE="AccntFileAutoUpdFlg" NAME="AccntFileAutoUpdFlg" />

   <FIELD VARIABLE="Comment" NAME="Comment" />

      </ROW>
```

NOTE: Remember that SWERowId=10-5O167A. Also, that in the XML output there is no data in the FIELD tags because this is a new form and the user has not entered any data.

TIP: After you upload the file, you can also verify that the filename is present in the XML file you get back from SWE.

6 Save the record by invoking the WriteRecord method with the value of the SWERowId parameter that you noted in the Step 5.

In the following example, Content-Type is application/x-www-form-urlencoded:

```
http://localhost/smc_enu/
start.swe?SWECmd=InvokeMethod&SWEApplet=Account+Attachment+Applet&SWEView=Account+
Attachment+View&SWERowId=10-5O167A&SWENeedContext=true&SWERowIds=SWERowId0=10-
5NIG6U&SWEReqRowId=1&SWEMethod=WriteRecord&SWEC=5&SWESetMarkup=XML
```

NOTE: Be sure to set the HTTP Content-Type to application/x-www-form-urlencoded for this step.

Downloading Files

Applications may require that files be downloaded from the Siebel database to a mobile device or other platform. For example, mobile voice applications may allow users to listen to voice recordings stored as audio files.

To download a file

1 Log in to SWE and navigate to the screen, view and applet to which you want to go.

For more information, see "Logging In" on page 35, "Navigating to a Screen" on page 36 and "Navigating Within a Screen" on page 37.

The Content-Type of the HTTP request must be set to application/x-www-form-urlencoded for each step of this procedure, except where noted.

2 Drill down in the record containing the attached file by invoking the Drilldown method and passing the value of the SWERowId parameter.

Example:

```
http://localhost/smc_enu/
start.swe?SWECmd=InvokeMethod&SWEApplet=Account+Attachment+Applet&SWEView=Account+
Attachment+View&SWERowId=10-5O167A&SWENeedContext=true&SWERowIds=SWERowId0=10-
5NIG6U,SWERowId1=&SWEReqRowId=1&SWEMethod=Drilldown&SWEC=3&SWEField=AccntFileName&
SWESetMarkup=XML
```

NOTE: SWERowID0 is the Row ID of the parent row and SWERowId is the child's ID. The Row IDs of the parent and child can be obtained from the XML output returned from the previous request. SWE requests that require a Row ID will not work if the SWERowId parameter is not supplied.

Getting Updates

In some instances, a third-party application may require a local copy of a subset of Siebel data for its own use. Such applications can periodically check if there is any new data, and retrieve just the data that has changed. The GetSMCUpdate method is used to obtain such updates. This method extracts the system level record information that is otherwise not available through the user interface.

NOTE: Generally, when using the Siebel Mobile Connector you can set the SWESetMarkup parameter to HTML, WML, or XML. However, you may not use the GetSMCUpdate method or receive alerts in any other markup than XML.

GetSMCUpdate

The method GetSMCUpdate is invoked to retrieve the changes in data from a specified time to the current time. The changes that are noted are at the record level versus for a specific field. Even if you apply a style sheet which parses the particular field that changes, the record will show as having changed. If your application requires notification about changes to specific fields, then you can use the Alert Business Service for this purpose. This is an important point for understanding this method.

The GetSMCUpdate method cannot be subscribed to at the Form Applet or Entry Applet level, only at the List Applet level. Also, if a field that has changed is present in both the List Applet and the Form Applet, then the GetSMCUpdate method will pick up the changes at the record level in the Form Applet.

When using the GetSMCUpdate method to retrieve updates for child applets, the method only returns the records for the child applet associated with a particular parent record.

NOTE: For a client application to receive updates, it is also necessary that the parameter VoiceApplication be set to TRUE. For more information, see "Configuring a User Agent for the Siebel Mobile Connector Applications" on page 25.

Send a command to SWE with the following parameters to get an update:

Parameters	Required or Optional	Description
SWECmd=InvokeMethod	Required	Sets the name of the SWE command to InvokeMethod.
SWEMethod=GetSMCUpdate	Required	Sets the name of the SWE method to GetSMCUpdate. This is a new SWE method available with the Siebel Mobile Connector.
SWEView=<view_name>	Required	Specifies the name of the view you want to receive an update for.
SWEApplet=<applet_name>	Required	Specifies the name of the applet you want to receive an update for. You must subscribe to the GetSMCUpdate at the List Applet level, not the Form Applet or Entry Applet level.
SWESetMarkup=XML	Required	Specifies the XML markup mode.
LastUpdate=<date>	Optional	Specifies the last update date or date/time. For example, 07/30/2002 or 07/30/2002 12:00:00. The time is denoted in 24 hour format (military time). The update is the delta between the current time and the time specified. If no value is specified, all requested data will be returned.

Parameters	Required or Optional	Description
SWEXslStyleSheet=<name_of_style_sheet>	Optional	Specifies an XSL style sheet to be applied to the output results.
SWESetRowCnt=<#_of_rows>	Optional	Specifies the number of rows to be applied. If no value is given, the number of rows returned matches the default value for NumberOfListRows specified in the smc.cfg file.

To get an update

1 Log in to SWE and navigate to the screen, view and applet to which you want to go.

For more information, see "Logging In" on page 35, "Navigating to a Screen" on page 36 and "Navigating Within a Screen" on page 37.

NOTE: It is not necessary to navigate to a specific view or applet if you want to get an update for the screen's default view.

2 Invoke the GetSMCUpdate method.

In the following example, you want to get an update for records that have changed from 07/01/2002 to the present:

```
http://localhost/sales/start.swe?
SWECmd=InvokeMethod&SWEMethod=GetSMCUpdate&SWEView=SMC+Account+View&SWEApplet=SMC+
Account+ListApplet&LastUpdate=07/01/2002&SWESetRowCnt=100&SWESetMarkup=XML
```

NOTE: If you pass an invalid date for the LastUpdate parameter, you may receive an error message Unable to load message 0xffff. This message indicates that the date given is not a valid parameter for the GetSMCUpdate method.

Additional Examples

This section lists several additional example requests to SWE using the GetSMCUpdate method.

The following request passes a date as the value of the LastUpdate parameter. It retrieves all records that have changed since 07/30/2002 or later.

```
http://localhost/smc_enu/
start.swe?SWECmd=InvokeMethod&SWEMethod=GetSMCUpdate&SWEView=SMC+Account+View&SWEA
pplet=SMC+Account+List+Applet&SWESetMarkup=XML&LastUpdate=07/30/2002
```

The following request passes a date and time as the value of the LastUpdate parameter. It retrieves all records that have changed since 07/30/2002 at noon or later.

```
http://localhost/smc_enu/
start.swe?SWECmd=InvokeMethod&SWEMethod=GetSMCUpdate&SWEView=SMC+Account+View&SWEA
pplet=SMC+Account+List+Applet&SWESetMarkup=XML&LastUpdate=07/30/2002 12:00:00
```

The following request passes a date as the value of the LastUpdate parameter and uses a style sheet to filter data.

It retrieves only the data specified in CompanyName_SiebelMobileConnector_SMCAccountView_GM.xsl for records that have changed since 07/30/2002 or later.

```
http://localhost/smc_enu/
start.swe?SWECmd=InvokeMethod&SWEMethod=GetSMCUpdate&SWEView=SMC+Account+View&SWEA
pplet=SMC+Account+List+Applet&SWESetMarkup=XML&LastUpdate=07/30/2002
12:00:00&SWEXslStyleSheet=CompanyName_SiebelMobileConnector_SMCAccountView_GM.xsl
```

NOTE: Remember that the GetSMCUpdate method retrieves changes at the record level and not for specific fields. If you apply a style sheet to limit data to a specific field, this will not make a difference. When your application must retrieve updates at the field level not the record level, then use the Alert Business Service.

The following request passes a date as the value of the LastUpdate parameter. It retrieves all records that have changed since 07/30/2002 or later, up to a maximum number of 50 records.

```
http://localhost/smc_enu/
start.swe?SWECmd=InvokeMethod&SWEMethod=GetSMCUpdate&SWEView=SMC+Account+View&SWEA
pplet=SMC+Account+List+Applet&SWESetMarkup=XML&LastUpdate=07/30/
2002&SWESetRowCnt=50
```

The following request passes a date as the value of the LastUpdate parameter. It retrieves all records that have changed since 07/30/2002 or later (assuming that there are less than 500 records).

```
http://localhost/smc_enu/
start.swe?SWECmd=InvokeMethod&SWEMethod=GetSMCUpdate&SWEView=SMC+Account+View&SWEA
pplet=SMC+Account+List+Applet&SWESetMarkup=XML&SWESetRowCnt=500
```

Global Language Support

Siebel applications are designed to meet the needs of customers operating in a global environment. The Siebel Mobile Connector API has an English interface that can be used to create applications in multiple languages to meet the needs of customers.

The management of different languages in third-party applications is a task that must be handled by the middleware application server and its communication to various devices.

For general information about deploying Siebel applications in a global environment, see *Siebel Global Deployment Guide*.

5 Working with the Metadata Business Service

The Metadata Business Service allows customers and partners to filter data offered through the preconfigured application definitions without having to use Siebel Tools to permanently change the application configuration. This section provides an overview of the Metadata Business Service, how it works, and its API. Additionally, you can learn about an application, the Reference Configuration Sample, that is a model for creating applications that access the Metadata Business Service API.

How the Metadata Business Service Works

Figure 5 illustrates how the Metadata Business Service is used to create style sheets and workflows.

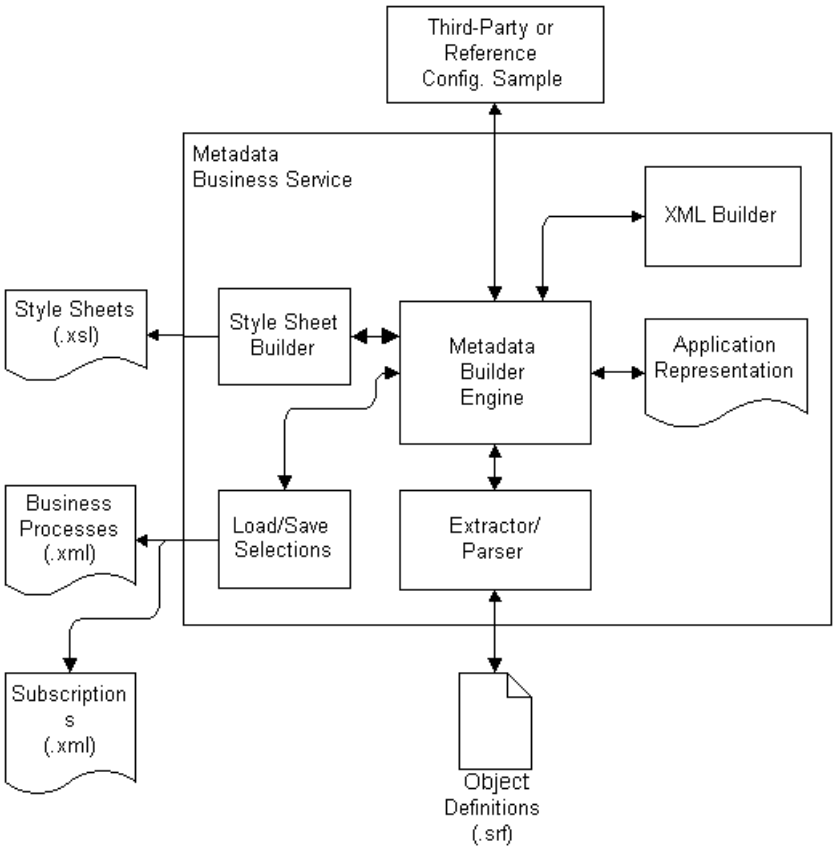


Figure 5. Metadata Business Service

Architecture Components

The following components are involved in the Metadata Business Service:

Metadata Builder Engine. This component receives input from and provides output to the third party or Reference Configuration Sample. It controls the extraction of the metadata from the SRF, the creation of an internal application representation, the building of XML representation for output, the building of style sheets and the loading and saving of XML subscriptions.

Third Party or Reference Configuration Sample. This sample is provided to show how to access the Metadata Business Service. The sample generates style sheets used by the third-party application and alerts based on defined business processes. It also allows the application developer to configure smcalert.cfg, the configuration file used by the Alert Business Service to determine the transport mechanism. It is hosted by an enterprise on a Microsoft Windows 2000 Web server. Third parties may wish to expose this capability within their own toolset, allowing developers to configure the third-party application.

XML Builder. This component is a module within the Metadata Business Service that traverses the hierarchical tree of the application representation and builds an XML representation for it.

Application Representation. This is a hierarchical tree structure of the object definitions obtained from the Extractor/Parser module. Each node in the hierarchical tree represents an object, such as a screen object, view object, applet object, and so forth.

Extractor/Parser. The Extractor/Parser is the module that traverses the application definition in the SRF and retrieves the necessary information about the user interface elements. It extracts only the user interface components necessary to create the XML output needed to configure the application definition (that is, the application, screens, views, applets, columns and list controls).

Style Sheet Builder. This component generates XSL style sheets conforming to the Siebel Web Engine DTD. The algorithm goes through each node of the application representation to check if the node is selected, and if so, the matching XML tag is copied.

Load/Save Selections. This component saves the XML subscription the user is working on to a file in the local file system. When the user loads the subscription from the file, the Metadata Builder engine also extracts a new representation from the SRF (in case there were any changes to the application definition) and merges the subscription with the internal application representation. Subscriptions and business processes are saved in the siebsrvr\xml directory at the location where Siebel Server is installed.

Metadata Process Flow

These steps describe the flow for creating new style sheets in greater detail.

1. Using the Reference Configuration Sample (or a a third-party user interface), the administrator establishes a connection with the Metadata Builder engine through a COM Data Control.
2. The administrator logs in to the Metadata Business Service and selects a Siebel application to view.

3. The Metadata Builder engine uses the Extractor/Parser functionality to extract and parse the object definitions (located in the SRF file) for the specified application. (The application definitions are defined in Siebel Tools.)
4. The Metadata Builder engine then creates a hierarchical representation of the entire application in memory.
5. The Metadata Builder engine uses the XML Builder functionality to generate an XML representation of the application. The XML representation is presented to the administrator in the Reference Configuration Sample as a hierarchical tree structure.
6. Using the Reference Configuration Sample, the administrator customizes the data in the hierarchical tree structure, selecting the user interface elements desired for each screen. (The administrator can also subscribe for updates and pushed alerts by using the same user interface.)
7. When the administrator has configured each screen of the application that needs to be customized, the administrator saves the selections as a subscription. The Metadata Builder engine uses the Save Selections functionality to save all the selections (user interface elements, updates, and alerts). If alerts have been subscribed for, the Metadata Builder engine creates business processes and saves them.
8. The administrator clicks the Create Stylesheets button. The Metadata Builder engine invokes the Style Sheet Builder functionality to generate the style sheets.

The Metadata Business Service API

Table 2 lists the API methods for accessing the Metadata Business Service. Third-party applications do not invoke the methods of the Metadata Business Service directly. Instead, the methods are invoked by calling the InvokeMethod method of the Siebel COM Data Control or Java Data Bean API. For more information about these APIs, see *Siebel Tools Online Help*.

Table 2. API Methods for Accessing the Metadata Business Service

Supported Values	Description	Arguments (with Description)
CanInvokeMethod	Returns whether or not the method can be invoked.	*MethodName* indicates the name of the method that you want to call.
InvokeMethod	Specifies a method to be invoked.	*MethodName* indicates the name of the method that you want to call. *inputArgs* are input arguments *outputArgs* are output arguments
CreateStyleSheets	Generates style sheets.	*inputArgs* are input arguments *outputArgs* are output arguments

Table 2. API Methods for Accessing the Metadata Business Service

Supported Values	Description	Arguments (with Description)
GetApplicationDef	Returns a hierarchical representation of the application definition.	*inputArgs* are input arguments *outputArgs* are output arguments
LoadSubscription	Loads a subscription from a file.	*inputArgs* are input arguments *outputArgs* are output arguments
MetadataCleanup	Destroys the created objects to free system resources.	*inputArgs* are input arguments *outputArgs* are output arguments
SaveSubscription	Saves a subscription to a file.	*inputArgs are input arguments* *outputArgs are output arguments*

There is a specific order that the methods are called in.

1. Call GetApplicationDef or LoadSubscription (to use LoadSubscription you must have previously used GetApplicationDef and saved a subscription).
2. Call SaveSubscription (can be called multiple times in one session).
3. Call CreateStyleSheets (can be called multiple times in one session).
4. Call MetadataCleanup.

NOTE: The Metadata Business Service is instantiated just like any other Siebel business service. You can instantiate the Metadata Business Service though the Siebel COM Data Control, the Java Data Bean Control, the XML Web Interface to SWE or any other object interface. For the COM and Java controls, you use the GetService(service_name) method on the application object. For the SWE interface, you use the LoadService(service_name) command.

CanInvokeMethod

CanInvokeMethod returns whether or not a method can be called.

Syntax

virtual BOOL CanInvokeMethod (const SSchar* pMethodName);

Argument	Description
MethodName	Indicates the name of the method that you want to call.

Returns

TRUE indicates that the method can be invoked; FALSE indicates otherwise.

Examples

```
CanInvokeMethod(SStext("SaveSubscription"));

CanInvokeMethod(SStext("GetApplicationDef"));
```

InvokeMethod

InvokeMethod calls a method.

Syntax

virtual ErrCode InvokeMethod (const SSchar* *pMethodName*,

const *CCFPropertySet* & *inputArgs*,

CCFPropertySet & *outputArgs*);

Argument	Description
MethodName	Indicates the name of the method that you want to call.
inputArgs	The input arguments of the method that you want to call.
outputArgs	The output arguments of the method that you want to call.

Returns

ErrCode. TRUE indicates success; FALSE indicates failure.

Example

```
InvokeMethod(SStext("CreateStyleSheets"), inputs, outputs);
```

CreateStyleSheets

CreateStyleSheets generates style sheets that can be passed to SWE to limit the data returned from the Siebel application. Do not directly call this method; instead call it from InvokeMethod.

Syntax

ErrCode CreateStyleSheets(const *CCFPropertySet* & *inputArgs*,

CCFPropertySet & outputArgs);

Argument	Description
inputArgs	The input arguments. <CREATE> SUBSCRIPTIONS='%string_of_XML%'. For a definition describing the inbound XML document, see "DTD for Subscriptions" on page 159.
outputArgs	The output arguments. <CREATE> SUCCESS=' '%success%'='TRUE' STYLESHEETS=' '%list_of_stylesheets_and_views%'

Returns

ErrCode. TRUE indicates success; FALSE indicates failure.

Example

```
InvokeMethod(SStext("CreateStyleSheets"), inputs, outputs);

The input argument should be in the following format:"

<?xml version="1.0" ?>

- <SUBSCRIPTIONS>

   <NODE NodeId="Siebel Mobile Connector" SubscribeForUI="FALSE"
   SubscribeForUpdate="FALSE" SubscribeForPush="FALSE" />

   <NODE NodeId="Siebel Mobile Connector_SMC Account Screen" SubscribeForUI="TRUE"
   SubscribeForUpdate="FALSE" SubscribeForPush="FALSE" />

...

<NODE NodeId="Siebel Mobile Connector_SMC Service Request Screen_Service Request
detail view w/attachments_Service Request Attachment List
Applet_ActivityFileName_ActivityFileName" SubscribeForUI="FALSE"
SubscribeForUpdate="FALSE" SubscribeForPush="FALSE" />

   </SUBSCRIPTIONS>
```

GetApplicationDef

GetApplicationDef returns a hierarchical representation of the application definition. Do not directly call this method; instead call it from the InvokeMethod.

Syntax

ErrCode GetApplicationDef (const *CCFPropertySet* & *inputArgs*,

CCFPropertySet & *outputArgs*);

Argument	Description
inputArgs	The input arguments. APPLICATION_NAME='%application_name%'
outputArgs	The output arguments. SUCCESS='%success%' = 'TRUE' XML_DEF='%string_of_XML%'. For a definition describing the outbound XML document, see "DTD for Application Definition" on page 155.

Returns

ErrCode. TRUE indicates success; FALSE indicates failure.

Example

```
InvokeMethod(SStext("GetApplicationDef"), inputs, outputs);

The following is an example of the output arguments:

<?xml version="1.0" encoding="UTF-16"?><?Siebel-Property-Set EscapeNames="false"?>

<APPLICATION NodeId="Siebel Mobile Connector" PushSelectValue="FALSE"
SubscribeForUpdate="FALSE" Name="Siebel Mobile Connector" SubscribeForUI="FALSE"
UpdateSelectValue="FALSE" UISelectValue="FALSE" SubscribeForPush="FALSE">

<SCREEN NodeId="Siebel Mobile Connector_SMC Account Screen" PushSelectValue="FALSE"
SubscribeForUpdate="TRUE" Default_View="SMC Account View" Name="SMC Account Screen"
Viewbar_Text="Account" SubscribeForUI="TRUE" UpdateSelectValue="FALSE"
UISelectValue="FALSE" SubscribeForPush="TRUE">

<VIEW NodeId="Siebel Mobile Connector_SMC Account Screen_SMC Account Detail -
Opportunities View" PushSelectValue="FALSE" SubscribeForUpdate="TRUE"
Business_Object="Account" Title="Account Opportunities"
Visibility_Type="SalesRepView:0" Name="SMC Account Detail - Opportunities View"
Viewbar_Text="Opportunities" SubscribeForUI="TRUE" UpdateSelectValue="FALSE"
UISelectValue="FALSE" SubscribeForPush="TRUE">

<APPLET NodeId="Siebel Mobile Connector_SMC Account Screen_SMC Account Detail -
Opportunities View_Account Form Applet" PushSelectValue="FALSE" Title="Account"
Business_Component="Account" Name="Account Form Applet" Base_Table="S_PARTY"
SubscribeForUI="TRUE" UISelectValue="FALSE" SubscribeForPush="TRUE"> <CONTROL
Table="S_ORG_EXT" Type="FT_EDIT" Pick_Applet="Account Status Pick Applet"
NodeId="Siebel Mobile Connector_SMC Account Screen_SMC Account Detail -
Opportunities View_Account Form  Applet_AccountStatus" PushSelectValue="FALSE"
Runtime="TRUE" Display_Name="Status" Name="AccountStatus" Field="Account Status"
SubscribeForUI="TRUE" UISelectValue="FALSE" SubscribeForPush="TRUE">
```

```
<FIELD Text_Length="30" Type="Text" Table="S_ORG_EXT" Extension="TRUE"
Calculated="FALSE" NodeId="Siebel Mobile Connector_SMC Account Screen_SMC Account
Detail - Opportunities View_Account Form  Applet_AccountStatus_Account Status"
PushSelectValue="FALSE" SubscribeForUpdate="FALSE" Multivalued="FALSE"
Name="Account Status" SubscribeForUI="FALSE" UpdateSelectValue="FALSE"
UISelectValue="FALSE" SubscribeForPush="FALSE">

<PICKLIST NodeId="Siebel Mobile Connector_SMC Account Screen_SMC Account Detail -
Opportunities View_Account Form  Applet_AccountStatus_Account Status_PickList
Account Status" PushSelectValue="FALSE" SubscribeForUpdate="FALSE"
Value="ACCOUNT_STATUS" Name="PickList Account Status" SubscribeForUI="FALSE"
UpdateSelectValue="FALSE" UISelectValue="FALSE" SubscribeForPush="FALSE"></
PICKLIST></FIELD></CONTROL>

<CONTROL Table="S_ORG_EXT" Type="FT_EDIT" NodeId="Siebel Mobile Connector_SMC
Account Screen_SMC Account Detail - Opportunities View_Account Form  Applet_Alias"
PushSelectValue="FALSE" Display_Name="Alias" Name="Alias" Field="Alias"
SubscribeForUI="TRUE" UISelectValue="FALSE" SubscribeForPush="TRUE">

<FIELD Text_Length="100" Type="Text" Table="S_ORG_EXT" Extension="TRUE"
Calculated="FALSE" NodeId="Siebel Mobile Connector_SMC Account Screen_SMC Account
Detail - Opportunities View_Account Form  Applet_Alias_Alias"
PushSelectValue="FALSE" SubscribeForUpdate="FALSE" Multivalued="FALSE" Name="Alias"
SubscribeForUI="FALSE" UpdateSelectValue="FALSE" UISelectValue="FALSE"
SubscribeForPush="FALSE"></FIELD></CONTROL>

<CONTROL Table="" Type="FT_BUTTON" SubscribeForUpdate="FALSE" NodeId="Siebel Mobile
Connector_SMC Account Screen_SMC Account Detail - Opportunities View_Account Form
Applet_EditRecord" PushSelectValue="FALSE" Display_Name="Edit" Name="EditRecord"
Field="" SubscribeForUI="TRUE" UISelectValue="FALSE" SubscribeForPush="FALSE"></
CONTROL>

<CONTROL Table="" Type="FT_BUTTON" SubscribeForUpdate="FALSE" NodeId="Siebel Mobile
Connector_SMC Account Screen_SMC Account Detail - Opportunities View_Account Form
Applet_GotoNextSet" PushSelectValue="FALSE" Display_Name="Next" Name="GotoNextSet"
Field="" SubscribeForUI="TRUE" UISelectValue="FALSE" SubscribeForPush="FALSE"> </
CONTROL>

<CONTROL Table="" Type="FT_BUTTON" SubscribeForUpdate="FALSE" NodeId="Siebel Mobile
Connector_SMC Account Screen_SMC Account Detail - Opportunities View_Account Form
Applet_GotoPreviousSet" PushSelectValue="FALSE" Display_Name="Previous"
Name="GotoPreviousSet" Field="" SubscribeForUI="TRUE" UISelectValue="FALSE"
SubscribeForPush="FALSE"></CONTROL></APPLET></VIEW></SCREEN></APPLICATION>
```

LoadSubscription

LoadSubscription gets the application definition and loads a subscription from a file. Subscriptions contain all the necessary metadata for generating style sheets or workflows. Do not directly call this method; instead call it from InvokeMethod.

Syntax

ErrCode LoadSubscription (const *CCFPropertySet* & *inputArgs*,

CCFPropertySet & *outputArgs*);

Argument	Description
inputArgs	The input arguments. <LOAD> FILE_NAME = '%full_path_to_filename%'. Pass the full path of a filename for an XML file conforming to the DTD for subscriptions. For a definition of the XML file, see "DTD for Subscriptions" on page 159.
outputArgs	The output arguments. <LOAD> SUCCESS=' '%success%'='TRUE' XML_DEF='%string_of_XML%'. For a definition describing the outbound XML document, see "DTD for Application Definition" on page 155.

Returns

ErrCode. TRUE indicates success; FALSE indicates failure.

Example

```
InvokeMethod(SStext("LoadSubscription"), inputs, outputs);

The following is an example of the output arguments:

<?xml version="1.0" encoding="UTF-16"?><?Siebel-Property-Set EscapeNames="false"?>

<APPLICATION NodeId="Siebel Mobile Connector" PushSelectValue="FALSE"
SubscribeForUpdate="FALSE" Name="Siebel Mobile Connector" SubscribeForUI="FALSE"
UpdateSelectValue="FALSE" UISelectValue="FALSE" SubscribeForPush="FALSE">

...

<CONTROL Table="" Type="FT_BUTTON" SubscribeForUpdate="FALSE" NodeId="Siebel Mobile
Connector_SMC Account Screen_SMC Account Detail - Opportunities View_Account Form
Applet_GotoPreviousSet" PushSelectValue="FALSE" Display_Name="Previous"
Name="GotoPreviousSet" Field="" SubscribeForUI="TRUE" UISelectValue="FALSE"
SubscribeForPush="FALSE"></CONTROL></APPLET></VIEW></SCREEN></APPLICATION>
```

MetadataCleanup

MetadataCleanup destroys the created objects to free system resources. You should call MetadataCleanup when you are finished with a session. Do not directly call this method; instead call it from InvokeMethod.

Syntax

void MetadataCleanup (const *CCFPropertySet* & *inputArgs*,

CCFPropertySet & *outputArgs*);

Argument	Description
inputArgs	The input arguments. Empty.
outputArgs	The output arguments. Empty.

Returns

Void

Example

```
InvokeMethod(SStext("MetadataCleanup"), inputs, outputs);

In Visual Basic Script:

Dim inputPropertySet, outputPropertySet

Set inputPropertySet = siebel.NewPropertySet()

Set outputPropertySet = siebel.NewPropertySet()

service_metadata.InvokeMethod("MetadataCleanup",
inputPropertySet,outputPropertySet)
```

SaveSubscription

SaveSubscription saves a subscription to a file. Subscriptions contain all the necessary metadata for generating style sheets or workflows. Do not directly call this method; instead call it from InvokeMethod.

Syntax

ErrCode SaveSubscription (const *CCFPropertySet* & *inputArgs*,

CCFPropertySet & outputArgs);

Argument	Description
inputArgs	The input arguments. <SAVE> SUBSCRIPTIONS='%string_of_XML%' FILE_NAME = '%full_path_to_filename%'. For a definition describing the inbound XML document, see "DTD for Subscriptions" on page 159.
outputArgs	The output arguments. <SAVE> SUCCESS=' '%success%'='TRUE'

Returns

ErrCode. TRUE indicates success; FALSE indicates failure.

Example

```
InvokeMethod(SStext("SaveSubscription"), inputs, outputs);

The input arguments should be in the following format:

<?xml version="1.0" ?>

- <SUBSCRIPTIONS>

   <NODE NodeId="Siebel Mobile Connector" SubscribeForUI="FALSE"
   SubscribeForUpdate="FALSE" SubscribeForPush="FALSE" />

   <NODE NodeId="Siebel Mobile Connector_SMC Account Screen" SubscribeForUI="TRUE"
   SubscribeForUpdate="FALSE" SubscribeForPush="FALSE" />

...

<NODE NodeId="Siebel Mobile Connector_SMC Service Request Screen_Service Request
detail view w/attachments_Service Request Attachment List
Applet_ActivityFileName_ActivityFileName" SubscribeForUI="FALSE"
SubscribeForUpdate="FALSE" SubscribeForPush="FALSE" />

   </SUBSCRIPTIONS>
```

Using the Reference Configuration Sample

The Reference Configuration Sample is an application that allows you to access the Siebel Mobile Connector Metadata Business Service API through a graphical user interface. You can use its graphical user interface to generate style sheets and alerts.

Performing Common Tasks

The Reference Configuration Sample simplifies common tasks that you will frequently perform with the Siebel Mobile Connector.

Table 3 lists the common tasks and the screens on which you perform them.

Table 3. Common Tasks

Common Task	Performed on This Screen
Choose an existing vendor name	Welcome screen (registration.asp)
Choose a new vendor name	Welcome screen (registration.asp)
Choose a Siebel application for which you want to create style sheets	Siebel applications screen (applications.asp)
Choose a Siebel application for which you want to create business processes	Siebel applications screen (applications.asp)
Log in to the application	Siebel applications screen (applications.asp)
Log off from the application	Metadata extractor screen (extractor.asp) or Style sheets screen (stylesuccess.asp)
Configure the alert transport method	SMC alert welcome screen (alertwelcome.asp)
Load a new subscription	Load subscription screen (load_file.asp)
Load an existing subscription	Load subscription screen (load_file.asp)
Choose a screen that you want to create subscriptions for	Metadata extractor screen (extractor.asp)
Generate style sheets	Metadata extractor screen (extractor.asp)
Extract a hierarchical representation of the XML data	Metadata extractor screen (extractor.asp)
Create or save subscriptions	Subscription configuration screen (view.asp)
Subscribe for style sheets to be applied to the user interface of the application definition	Subscription configuration screen (view.asp)
Subscribe for style sheets to be applied for updates	Subscription configuration screen (view.asp)
Subscribe for workflows that provide alerts	Subscription configuration screen (view.asp)
View generated style sheets	Style sheets screen (stylesuccess.asp)
View generated workflows	Style sheets screen (stylesuccess.asp)

NOTE: If you receive an error message from the Siebel Server while using the Reference Configuration Sample, close the browser and reopen a new browser to avoid receiving incorrect error messages. If you do not close and reopen a new browser, the error message will continue to be displayed even if the operation you are doing is valid.

Applying Style Sheets

You can limit the Siebel data returned to the third-party application using the XSL style sheets generated by the Reference Configuration Sample. The XML query parameter SWEXslStyleSheet allows you to pass the name of the style sheet you want to apply to the XML output. The command must be in the format of SWEXslStyleSheet=<style sheet name>.

In this example, Siebel data from the SMC Account List Applet is limited by applying a style sheet called CompanyName_SiebelMobileConnector_SMCAccountView_GM.xsl.

```
http://localhost/smc_enu/
start.swe?SWECmd=InvokeMethod&SWEMethod=GetSMCUpdate&SWEView=SMC+Account+View&SWEApple
t=SMC+Account+List+Applet&SWESetMarkup=XML&LastUpdate=07/30/2002
12:00:00&SWEXslStyleSheet=CompanyName_SiebelMobileConnector_SMCAccountView_GM.xsl
```

TIP: Apply style sheets to an application in order to limit data to the minimum necessary so that the application can generate its output. This will improve performance by minimizing the data sent back to the third-party application. For example, a mobile voice application should receive only enough data required to generate its speech grammars.

For additional information about using XSL style sheets with SWE, see *Siebel Tools Online Help*.

Reference Configuration Sample User Interface

This section provides an overview of commonly used screens provided in the Reference Configuration Sample user interface.

Welcome Screen

You can perform the following task on the welcome screen (registration.asp):

Enter a new vendor name or choose from a selection of names that you have previously provided. The vendor name you give here will be used as part of the filename for XML and XSL files generated by the Reference Configuration Sample. For example, if you enter CompanyName for the vendor name, a sample subscription file will be named CompanyName_SiebelMobileConnector.xml.

NOTE: The company name is meant to be unique in order to differentiate style sheets when creating them. If you choose to have multiple style sheets for a specific service, then you will need to create another unique name.

Figure 6 is an example of the Welcome Screen (registration.asp). When you have entered the vendor name, click Submit to go to the next screen.

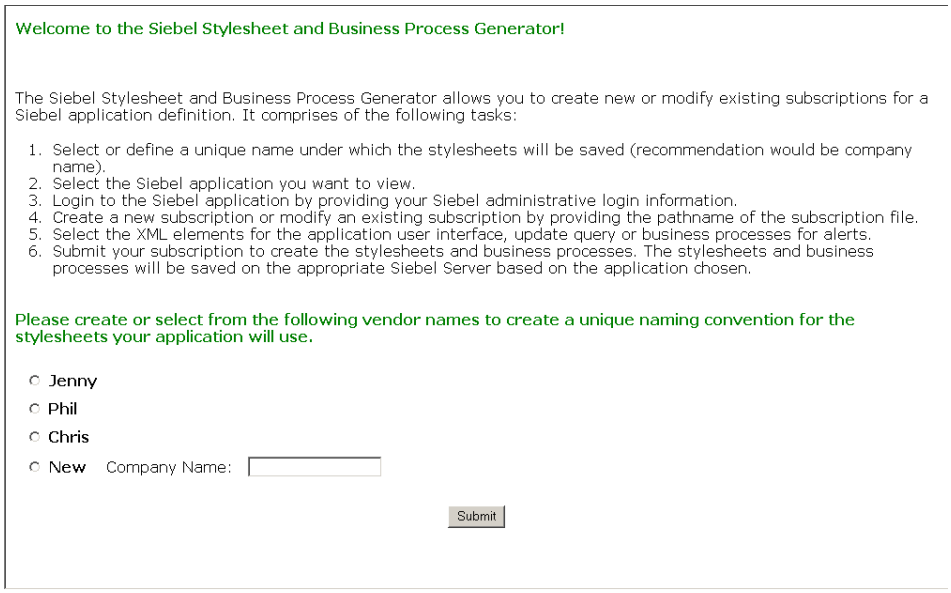


Figure 6. Welcome Screen (registration.asp)

Siebel Applications Screen

You can perform the following tasks on the Siebel applications screen (applications.asp):

Enter the Siebel application for which you want to create style sheets or business processes. A list of Siebel applications appears on this screen based on the contents of the metadata.cfg file. You can choose an application from the list or add a new application name and its object manager in the user interface.

Enter the log in name and password for the Siebel application to which you want to log in.

Figure 7 is an example of the Siebel applications screen (applications.asp). When you have selected the Siebel application and provided the log in user data, click Submit to go to the next screen.

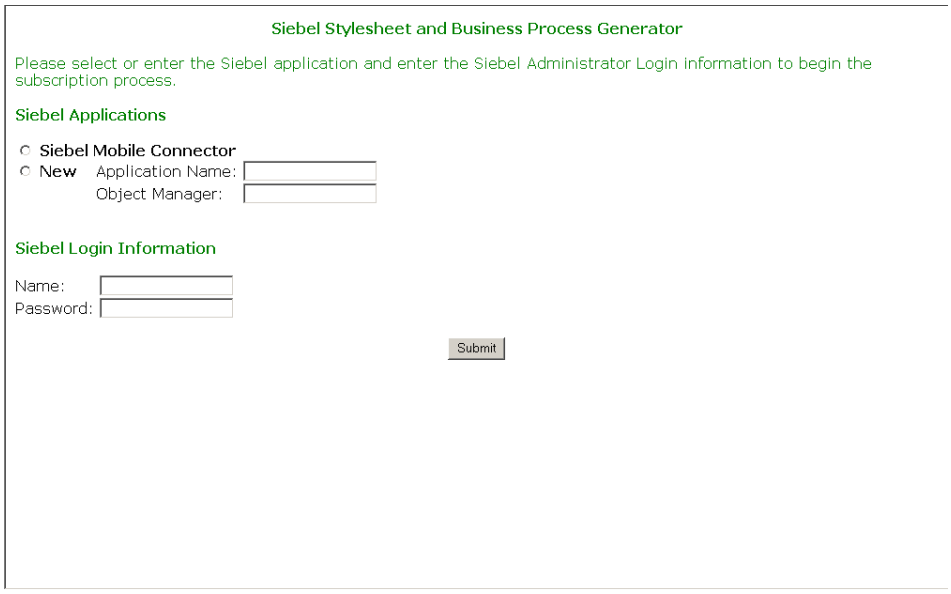


Figure 7. Siebel Applications Screen (applications.asp)

SMC Alert Welcome Screen

Use the SMC alert welcome screen (alertwelcome.asp) to specify the default transport method to be used with the Alert Business Service. If you choose to do this, click the Subscribe for Alerts button to modify the smcalert.cfg file.

Figure 8 is an example of the SMC alert welcome screen (alertwelcome.asp). To continue to the next screen, click Cancel Alerts.

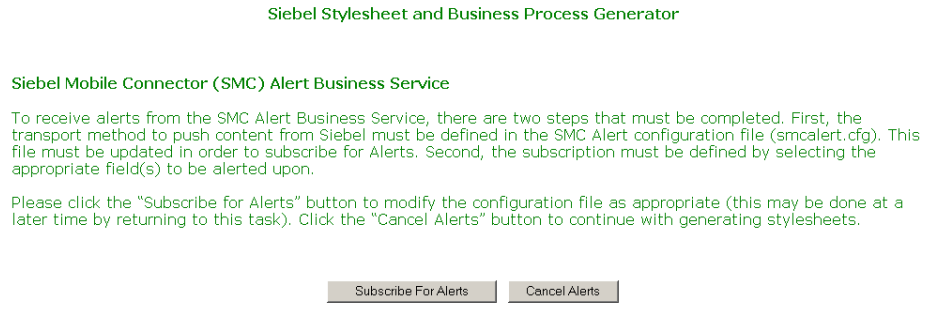


Figure 8. SMC Alert Welcome Screen (alertwelcome.asp)

Load Subscription Screen

Use the load subscription screen (load_file.asp) to specify whether you want to work with a new or existing subscription.

If you want to work with:

An existing subscription, select the Existing option and browse to locate the subscription file you want to load.

A new subscription, select the New option.

NOTE: If you change the application definition in Siebel Tools, subscriptions created and saved using older application definitions are not automatically updated to the new SRF. To make subscriptions reflect changes in the SRF, you must rerun the metadata extractor and create new subscriptions.

Figure 9 is an example of the Load subscription screen (load_file.asp). To continue to the next screen, click Submit.

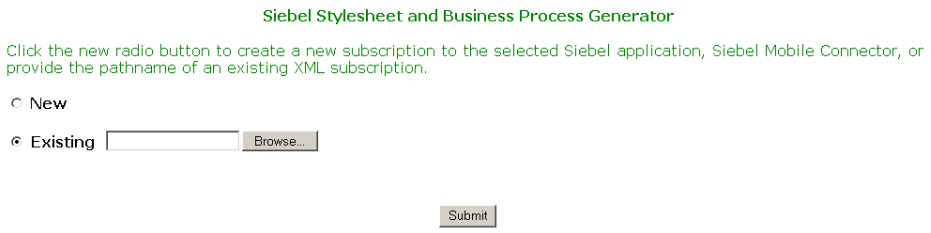


Figure 9. Load Subscription Screen (load_file.asp)

Metadata Extractor Screen

You can perform the following tasks on the metadata extractor screen (extractor.asp):

Choose a screen in the selected Siebel application that you want to create subscriptions for. You can click on a link to extract an XML representation of a screen in the application and view a hierarchical representation of the XML data.

Configure the alert transport method (if you skipped the alert configuration step earlier or if you want to update your current configuration).

Generate the style sheets in the subscription. You can click the Create Style sheets button to generate the style sheets in the subscription and navigate them (this option only appears after you have saved a subscription).

NOTE: Style sheets are saved in the siebsrvr\webtempl directory at the location where Siebel Server is installed. All style sheets are named beginning with *vendor name* where *vendor name* is the vendor name selected on the welcome screen (registration.asp).

Figure 10 is an example of the Metadata Extractor Screen (extractor.asp) before Style sheets are created.

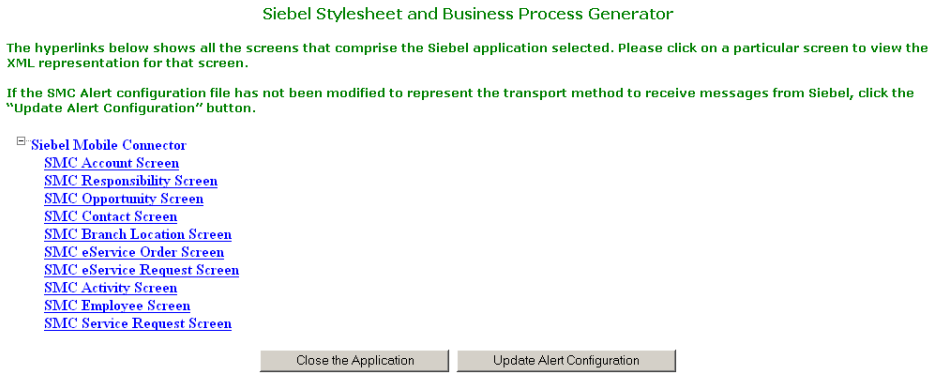


Figure 10. Metadata Extractor Screen (extractor.asp) Before Style Sheets Are Created

Generally, one style sheet is created for each view for which you have subscribed. Additionally, a style sheet is created for unsubscribed views. This style sheet is named *vendor name_application name*_UnsubscribedView_UI.xsl. When you apply this style sheet, all user interface elements and updates not explicitly subscribed for will not be returned from SWE.

Log off from SWE and terminate the browser session. You can do this by clicking Close Application.

NOTE: It is recommended that you log off from the Reference Configuration Sample application by clicking Close the Application on the user interface. By doing so, you will free up the objects created during a session of using the Metadata Business Service.

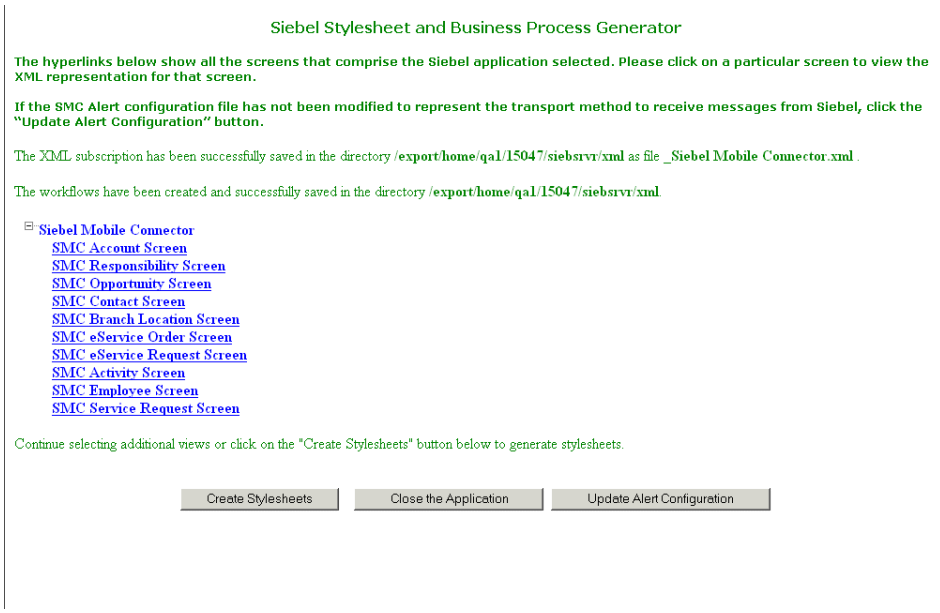


Figure 11. Metadata Extractor Screen (extractor.asp) After Style Sheets Are Created

Subscription Configuration Screen

The subscription configuration screen (view.asp) displays a hierarchical representation of the application definition for the Siebel application you have selected. This representation only displays the user interface layer elements configured in Siebel Tools. By default, there are no filters applied to a screen until selections have been defined on this screen.

You can perform the following tasks on this screen:

Subscribe for style sheets to be applied when retrieving a specific view of the application definition for the Siebel application. Select the Subscribe For User Interface check box for each screen, view, or applet element for which you want to subscribe.

Subscribe for style sheets to be applied specifically for updates (that is, calls to the GetSMCUpdate method). Select the Subscribe For Updates check box for each screen, view, or applet element for which you want to subscribe.

NOTE: SaveSubscription requires four base business processes to be present in the siebsrvr\xml directory; otherwise the Alert Business Service functionality will not work. These workflows are SMCAlert-Insert, SMCAlert-Delete, SMCAlert-NewValue and SMCAlert-OldValue.

Subscribe for workflows that provide alerts through the Alert (Push) Service. Select the Subscribe for Alert (Push) Service check box for each screen, view, or applet element for which you want to subscribe.

NOTE: Subscriptions are saved in the siebsrvr\xml directory at the location where Siebel Server is installed. Files are named following this convention: Vendor Name_Application Name.xml.

Figure 12 is an example of the Subscription configuration screen (view.asp). When you are done making your selections, you can save or cancel your selections by clicking Save Subscriptions and Return to Screens or Cancel and Return to Screens.

Figure 12. Subscription Configuration Screen (view.asp)

Style Sheets Screen

Use the style sheets screen (stylesuccess.asp), shown in Figure 13, to view the style sheets generated from the subscription. When you are done, you can go back to the previous page by clicking Return to Screens or you can log off from SWE and terminate the browser session by clicking Close the Application.

NOTE: It is important to log off from the application by clicking Close the Application on the user interface. Do not terminate your session by closing the browser window; if you do this, the SWE user session will not be terminated.

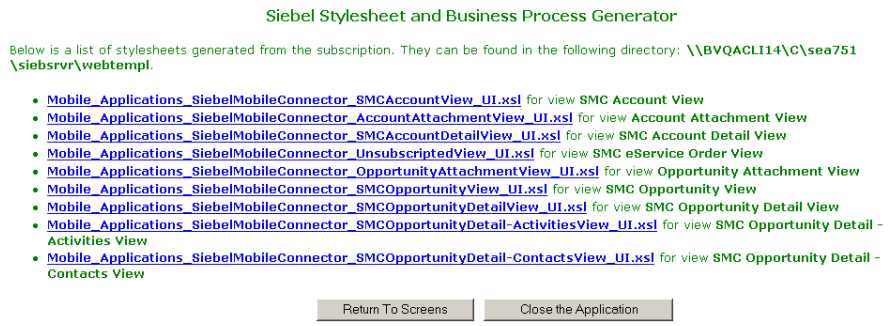


Figure 13. Style Sheets Screen (extractor.asp)

6 Working with the Alert Business Service

The Alert Business Service can push information in XML format to partner applications to notify them that the contents of requested views and applets have changed. For example, an alert could indicate that a service request has been reassigned from one technician to another. The partner application is responsible for processing the alerts from the Siebel Mobile Connector and transmitting them to the end-user.

How the Alert Business Service Works

Figure 14 illustrates how the Alert Business Service works within the Siebel system and delivers information to a third-party application.

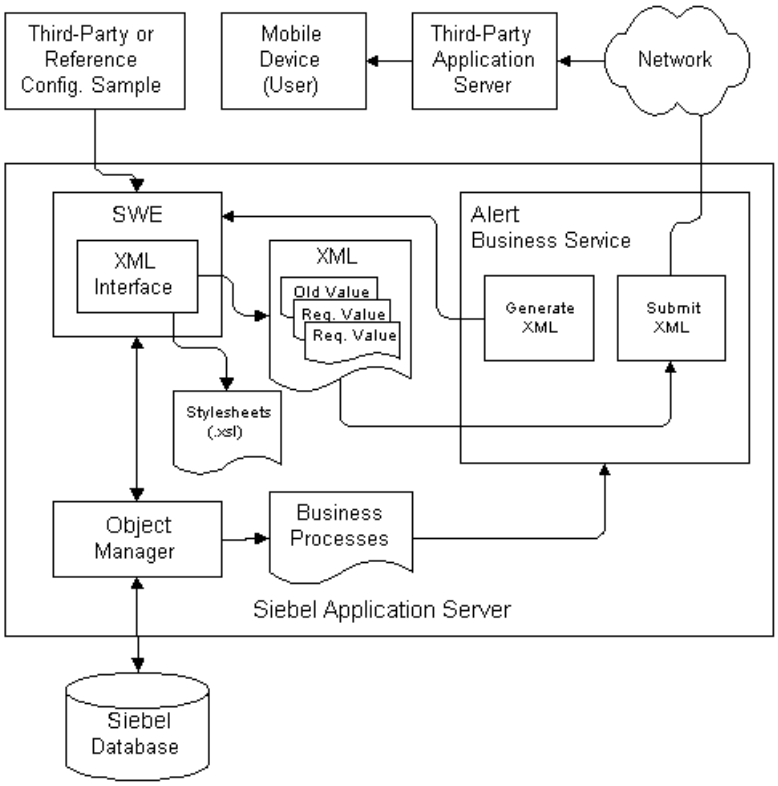


Figure 14. Alert Business Service

Alert Process Flow

These steps describe the flow in greater detail:

1. Using the Reference Configuration Sample, the administrator subscribes for alerts that will send notification messages to the third-party application when requested information has changed.
2. The Metadata Business Service creates subscriptions and generates workflow files based on the updates the administrator has subscribed for. The XML documents are saved in the siebsrvr\xml directory at the location where Siebel Server is installed.
3. The administrator imports the workflows into the Siebel Business Process Administration application and activates them. Once activated, the workflows monitor for specific data changes in the Siebel database until they are deactivated or deleted.
4. The Alert Business Service stores the old (prewrite) and new (postwrite) data of the specified business component and generates XML documents that contain both the old and new requested data when an alert condition is triggered.
5. The Alert Business Service submits the alert to a third-party application through a specified Siebel transport method. The supported transport methods include HTTP and MSMQ. For more information on supported transport mechanisms, see *Transports and Interfaces: Siebel Enterprise Application Integration*.
6. The third-party application receives the alert and processes the information. The application can use both the prewrite values and the postwrite values to determine which end users require the update.

Configuring Alerts

Before you can use the Alert Business Service, you must specify a transport mechanism, as described in "Configuring the smcalert.cfg File" on page 24. This section provides instructions and an example for configuring alerts.

NOTE: You can also create alerts directly with the Siebel Business Process Administration Screen. For more information, see *Siebel Business Process Framework: Workflow Guide*.

To import and activate a workflow

1. Using the Reference Configuration Sample, subscribe to an alert and save your subscription.

   For example, using the Reference Configuration Sample, check the Subscribe for Alert (Push) Service checkbox on the subscription configuration screen for a screen that contains the field you want to receive an alert for.
2. Generate the workflows based on your subscriptions.
3. Using the Siebel application, navigate to Site Map > Business Process Administration > Workflow Processes.
4. From the menu, select the Import Workflow option.

5 Browse to the directory where the Siebel Mobile Connector XML workflow file is located (for example, CompanyName_Account_Account_Name_NewValue).

6 Using the Workflow Designer, double-click on the Submit Message option (the third step).

7 Change the argument for Alert Transport to EAI HTTP Transport and save your changes.

8 From the menu, select Activate.

To deactivate a workflow

Navigate to the workflow in the Siebel application and select it. Then choose Deactivate from the menu.

Sample Alert Configuration Scenario

This section provides the following examples for configuring alerts:

Setting Up Alerts Using the HTTP Transport Method on page 75

Printing the Output Page of the Alert to a File on page 76

This scenario assumes you have already successfully configured the smcalert.cfg file.

Setting Up Alerts Using the HTTP Transport Method

The following example describes how to configure alerts using the HTTP Transport Method.

To set up alerts using the HTTP Transport Method

1 Create a physical directory on the Siebel Server that will receive the pushed alerts, such as D:\HTTPOutbound (where D:\ is a physical directory on the Siebel Server).

2 Create a program (for example, FileReceiver.asp) that receives an XML document and writes the output to a text file.

The program should point to the directory created in Step 1 and the name of the output text file. It must determine the character set and HTTP Content-Type, convert the character set to Unicode, and write the received document to a file.

TIP: You can model your program on the sample FileReceiver.asp shown in Appendix B, "Siebel Mobile Connector Troubleshooting Guide."

3 On the machine where you have installed the Reference Configuration Sample or similar third-party application, create a virtual directory and point the virtual directory to the HTTPOutbound physical folder created in Step 1.

4 Using the Reference Configuration Sample or a similar third-party application, subscribe to an alert for a specific list column or control.

For example, you can request to be alerted when the main phone number for an account has changed.

5 Using the Siebel application, navigate to Business Process Administration > Workflow Processes, and query for the imported workflow.

For example, you can run a query such as *SMC*.

6 From the menu, select the Import Workflow option and browse to the directory where the Siebel Mobile Connector XML workflow file is located.

For example, go to *machineName*\sea75x\siebsrvr\xml and import the appropriate XML file.

7 Using the Workflow Designer, double-click on the Submit Message option (the third step).

8 Change the argument for Alert Transport to EAI HTTP Transport and save your changes.

9 Activate the workflow.

10 Navigate to the smcalert.cfg file and add the following arguments:

```
[EAI HTTP Transport]

HTTPRequestURLTemplate = http://machineName/HTTPOutbound/FileReceiver.asp

 HTTPRequestMethod=POST
```

11 Save your changes to smcalert.cfg and then stop and restart the Siebel Server.

Printing the Output Page of the Alert to a File

Once the alert is configured, the following example describes how to print the output page to a file.

To print the output page of the alert to a file

1 Log in to the Siebel Mobile Connector client user interface.

2 Navigate to the screen containing the information for which you have configured an alert. For example, if you set up an alert for a change in an account's phone number, you can navigate to the SMC Account Screen.

3 Modify the value of the field and save your changes.

In the background, the Alert Business Service detects the change and sends a Siebel message through the HTTP transport mechanism.

4 Navigate to the location of the text output file.

For example: D:\HTTPOutbound\httpoutputfile.txt

5 Open the text file and you should see the XML document output from the Alert Business Service.

For example, if you requested to be notified when the main phone number for an account changed, httpoutputfile.txt would look like the following:

```
Received at 6/19/2002 9:54:10 AM

~< ? x m l  v e r s i o n = " 1 . 0 "  e n c o d i n g = " U T F - 8 " ? > < ? S i e b e l - P r o p e r t y - S e t  E s c a p e N a m e s = " f a l s e " ? > < P r o p e r t y S e t
```

```
AlertRowId="99-28B0A"

AlertDescription="SMC Alert - New Account Main Phone Number"

AlertMainPhoneNumber="4255551122"

></PropertySet>

End Received at 6/19/2002 9:54:10 AM
```

6 Restart the Siebel Server.

For more information about how the Alert Business Service works, see *Siebel Business Process Framework: Workflow Guide.*

A SMC Application Definition Quick Reference

This appendix describes the Siebel Mobile Connector (SMC) application definition with the corresponding objects and properties settings. It includes descriptions of the screens and views of the Siebel Mobile Connector application definition. The appendix also lists the applets along with the corresponding field names.

SMC Account Screen

Information relating to your accounts is recorded and tracked in the views of the SMC Account screen, as shown in Figure 15. An account represents the relationship between your company and the companies or individuals with whom you do business.

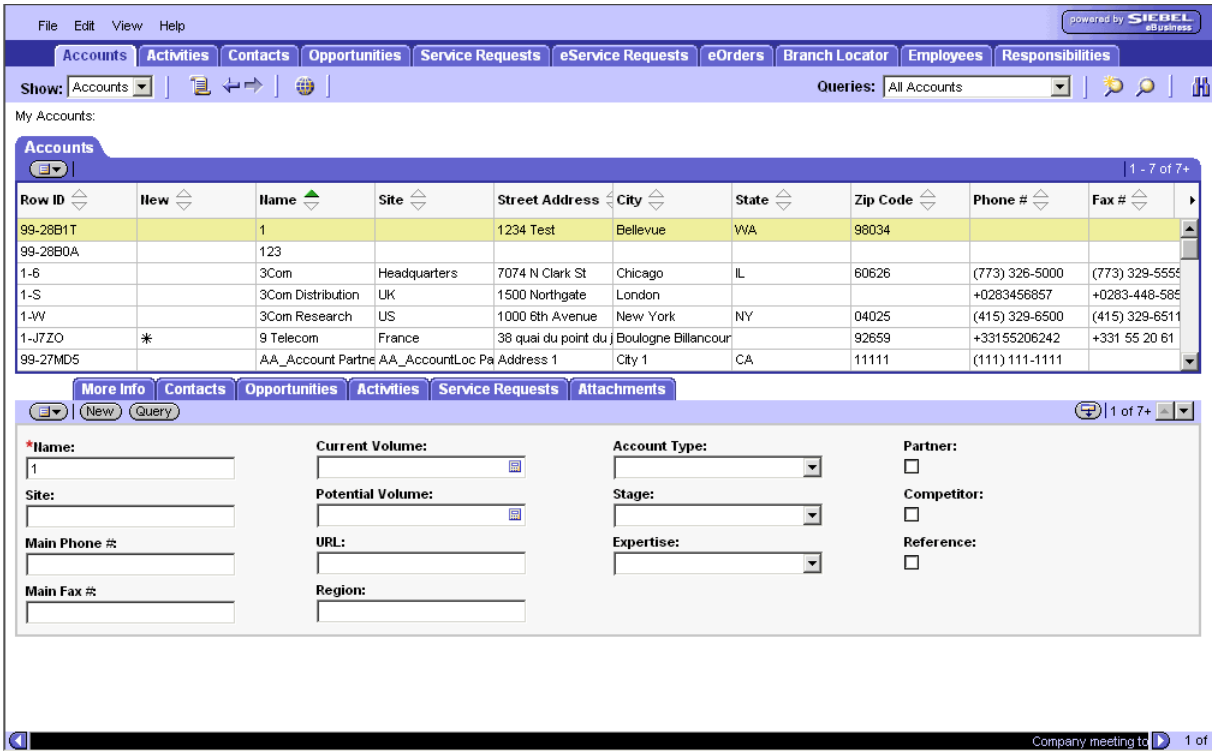


Figure 15. SMC Account Screen

The following lists the details for the SMC Account screen.

Display Name	Accounts
Business Object	Account
Business Component	Account

The following views are available from the SMC Account screen.

View Name	Viewbar Text	Sequence
SMC Account View	Accounts	1
SMC Account Detail View	More Info	2
SMC Account Detail - Contacts View	Contacts	3
SMC Account Detail - Opportunities View	Opportunities	4
SMC Account Detail - Activities View	Activities	5
SMC Account Detail - Service Requests View	Service Requests	6
Account Attachment View	Attachments	7

SMC Account View

The following lists the view details.

Screen	SMC Account Screen
View	SMC Account View
Business Object	Account
Applet	SMC Account List Applet, Account Form Applet
Visibility	Sales Rep

SMC Account Detail View

The following lists the view details.

Screen	SMC Account Screen
View	SMC Account Detail View
Business Object	Account
Applet	SMC Account List Applet, Account Form Applet
Visibility	Sales Rep

SMC Account Detail - Contacts View

The following lists the view details.

Screen	SMC Account Screen
View	SMC Account Detail - Contacts View
Business Object	Account
Applet	SMC Account Contact List Applet, Account Form Applet
Visibility	Sales Rep

SMC Account Detail - Opportunities View

The following lists the view details.

Screen	SMC Account Screen
View	SMC Account Detail - Opportunities View
Business Object	Account
Applet	SMC Oppty List Applet, Account Form Applet
Visibility	Sales Rep

SMC Account Detail - Activities View

The following lists the view details.

Screen	SMC Account Screen
View	SMC Account Detail - Activities View
Business Object	Account
Applet	SMC Account Activity List Applet, Account Form Applet
Visibility	Sales Rep

SMC Account Detail - Service Requests View

The following lists the view details.

Screen	SMC Account Screen
View	SMC Account Detail - Service Requests View
Business Object	Account
Applet	SMC Account Service List Applet, Activity Form Applet
Visibility	Sales Rep

Account Attachment View

The following lists the view details.

Screen	SMC Account Screen
View	SMC Account Attachment View
Business Object	Account
Applet	Account Entry Applet, Account Attachment View
Visibility	Sales Rep

SMC Account List Applet

The following lists the applet's details.

Applet Name	SMC Account List Applet
Business Component	Account

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Row ID	Id
New	Row Status
Name	Name
Site	Location
Street Address	Street Address
City	City
State	State
Zip Code	Postal Code
Phone #	Main Phone Number
Fax #	Main Fax Number
Status	Account Status
Industries	Industry
Currency Code	Currency Code
Price List	Price List
Sales Team	Sales Rep

Display Name	Field Name
Current Volume	Current Volume
Potential Volume	Total Potential Volume

Account Form Applet

The following lists the applet's details.

Applet Name Account Form Applet
Business Component Account

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Name	Name
Alias	Alias
Address	Street Address
City	Primary Account City
Site	Location
Zip Code	Primary Ship to Postal Code
State	State
Country	Primary Account Country
Region	Region
Type	Type
Status	Account Status
Main Phone #	Main Phone Number
Main Fax #	Main Fax Number
Parent	Parent Account Name
URL	Home Page
PO Approved	Competitor

SMC Account Contact List Applet

The following lists the applet's details.

Applet Name SMC Account Contact List Applet
Business Component Contact

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Last Name	Last Name
First Name	First Name
Middle Name	Middle Name
Work Phone #	Work Phone #
Job Title	Job Title
Email	Email Address
Account	Account
Site	Account Location
Employee Flag	Employee Flag
New	Row Status
Mr/Mrs	M/M
Registration Source	Registration Source App Name
Address Line 1	Street Address
Mobile Phone #	Cellular Phone #
City	City
State	State
Zip Code	Postal Code
Country	Country
Comments	Comment
Home Phone #	Home Phone #
Work Fax #	Fax Phone #
Households	Households
Alias	Alias
Survey Type	Account Survey Type

Display Name	Field Name
Survey Flag	Survey Flag
Survey Language	Survey Language
Account Id	Calculated Account Id

SMC Oppty List Applet

The following lists the applet's details.

Applet Name SMC Oppty List Applet
Business Component Opportunity

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Name	Name
Account	Account
Primary	Primary Sales Rep Login
Revenue	Revenue
Sales Stage	Sales Stage
Close Date	Primary Revenue Close Date
New	Row Status
Priority Flag	Priority Flag
Status	Status
Site	Account Location
Committed	Primary Revenue Committed Flag
Expected Value	Expected Value
Revenue Class	Primary Revenue Class
Probability %	Primary Revenue Win Probability
Channel	Channel
Description	Description
Lead Quality	Lead Quality
Reason	Reason Won Lost

Display Name	Field Name
Revenue Type	Primary Revenue Type
Sales Method	Sales Method
Source	Source
Lead Partner	Partner
Executive Priority	Executive Priority Flag
Executive Priority Date	Executive Priority Date

SMC Account Activity List Applet

The following lists the applet's details.

Applet Name SMC Account Activity List Applet
Business Component Action

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
New	Row Status
Description	Description
Type	Type
Start	Planned
Due	Due
Status	Status
Priority	Priority
Owner	Primary Owned By
Account	Account Name
Display In	Display
Site	Account Location
Opportunity	Opportunity
Activity #	Id
Private	Private

Display Name	Field Name
Parent Activity	Previous Activity Description
Comments	Comment
Meeting Location	Meeting Location
End	Planned Completion
Repeat Frequency	Repeating Type
Repeat Until	Repeating Expires
Alarm	Alarm
Attachments	(for future use)
Created	Created
Created By	Created By Name
Actual End	Done
Done	Done Flag
Employees	Owned By
Contacts	Contact Last Name

SMC Account Service List Applet

The following lists the applet's details.

Applet Name SMC Account Service List Applet
Business Component Service Request

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
New	Row Status
SR #	SR Number
Summary	Abstract
Account	Account
Site	Account Location
Product	Product

Display Name	Field Name
Priority	Priority
Status	Status
Substatus	Sub-Status
Area	Area
Subarea	Sub-Area
Severity	Severity
Owner	Owner
Entitlement Name	Entitlement Name
Commit Time	Commit Time
Date Closed	Closed Date
Contact First Name	Contact First Name
Contact Last Name	Contact Last Name
Organization	Organization
Date Opened	Created
Created By	Created By Name

Account Attachment Applet

The following lists the applet's details.

Applet Name Account Attachment Applet
Business Component Account Attachment

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Attachment Name	Accnt FileName
Size (In Bytes)	Accnt FileSize
Type	Accnt FileExt
Modified	Accnt FileDate
Update File	Accnt FileAutoUpdFlg
Comments	Accnt Comment

Account Entry Applet

The following lists the applet's details.

Applet Name Account Entry Applet
Business Component Account

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Name	Name
Site	Location
Main Phone #	Main Phone Number
Main Fax #	Main Fax Number
Address Line 1	Street Address
Address Line 2	Street Address 2
City	City
State	State
Zip Code	Postal Code
Country	Country
Rate	Rate
Organization	Organization
Current Volume	Current Volume
Potential Volume	Total Potential Volume
URL	Home Page
Region	Region
Account Team	Sales Rep
Parent	Parent Account Name
Parent Site	Parent Account Location
DUNS #:	DUNS Number
Domestic Ultimate DUNS	Domestic Ultimate DUNS
Parent/HQ DUNS	Parent HQ DUNS
Global Ultimate DUNS	Global Ultimate DUNS
Synonyms	Synonym

Display Name	Field Name
Account Type	Type
Stage	Reference Stage
Expertise	Expertise
Assignment Area Code	Assignment Area Code
Assignment Country Code	Assignment Country Code
Industries	Industry
Territories	Territory
Shipping Information	Freight Terms Info
Shipping Terms	Freight Terms
Inventory Location	Primary Fulfillment Inventory Location
Location Type	Location Type
Partner	Parent Account Name
Competitor	Competitor
Reference	Reference Flag
PO Approved	PO Approved Flag
Lock Assignment	Assignment Excluded
Disable Cleansing	Disable DataCleansing
Referenceable as of	Reference Date
Price List	Price List
Currency	Currency Code
Global Owner	S-S Instance
PO Auto-Approval Limit	PO Auto Approval Limit
Survey Type	Survey Type

SMC Activity Screen

Information relating to activities is recorded and tracked in the views of the SMC Opportunities screen, as shown in Figure 16. Activities are various tasks or events that are performed for contacts, accounts, and opportunities.

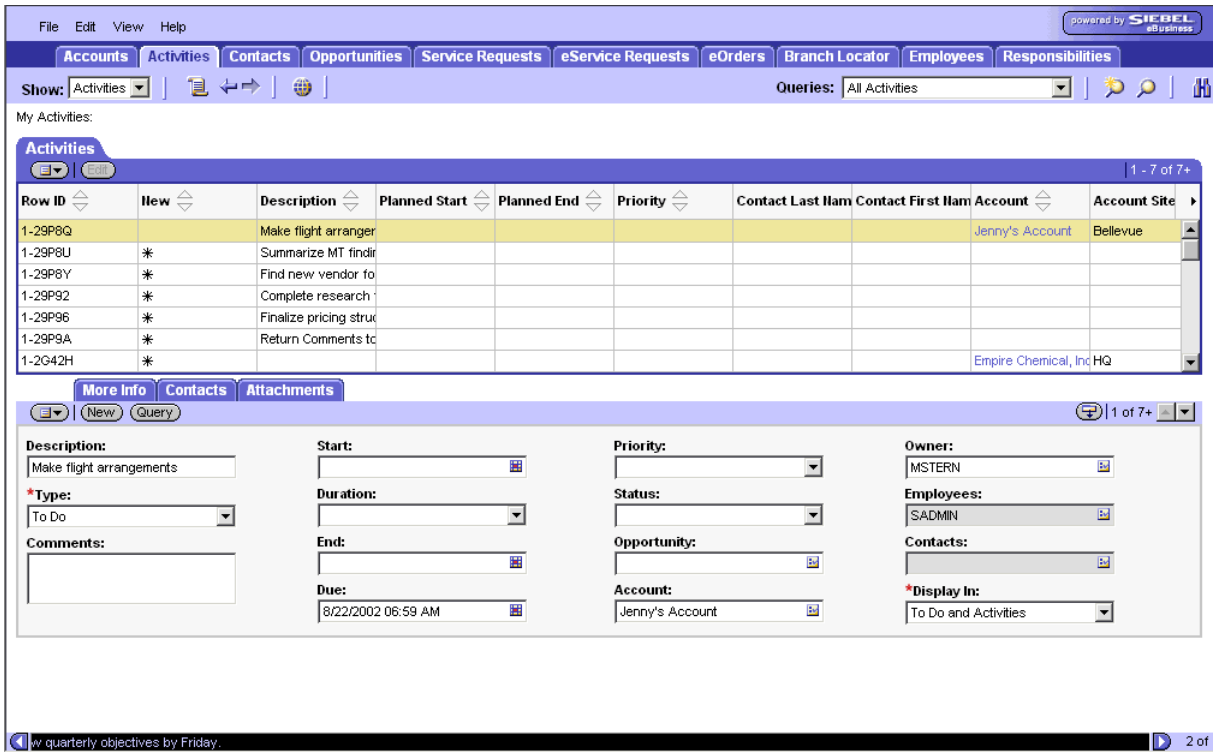


Figure 16. SMC Activity Screen

The following lists the details for the SMC Activity screen.

Display Name Activities
Business Object Action
Business Component Action

The following views are available from the SMC Activity screen.

View Name	Viewbar Text	Sequence
SMC Activity View	Activities	1
SMC Account Detail View	More Info	2

View Name	Viewbar Text	Sequence
SMC Activity Detail - Contacts View	Contacts	3
Activity Attachment View	Attachments	4

SMC Activity View

The following lists the view details.

Screen	SMC Activity Screen
View	SMC Activity View
Business Object	Action
Applet	SMC Activity List Applet, Activity Form Applet
Visibility	Personal

SMC Activity Detail View

The following lists the view details.

Screen	SMC Activity Screen
View	SMC Activity Detail View
Business Object	Action
Applet	SMC Activity List Applet, Activity Form Applet
Visibility	Personal

SMC Activity Detail - Contacts View

The following lists the view details.

Screen	SMC Activity Screen
View	Activity Detail - Contacts View
Business Object	Action
Applet	Activity Form Applet, SMC Activity Contact List Applet
Visibility	Personal

Activity Attachment View

The following lists the view details.

Screen	SMC Activity Screen
View	Activity Attachment View
Business Object	Action
Applet	Activity Form Applet, Activity Attachment Applet
Visibility	Personal

SMC Activity List Applet

The following lists the applet's details.

Applet Name	SMC Activity List Applet
Business Component	Action

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Row ID	Id
New	Row Status
Description	Description
Planned Start	Planned
Planned End	Planned Completion
Priority	Priority
Contact Last Name	Contact Last Name
Contact First Name	Contact First Name
Account	Account Name
Account Site	Account Location
Opportunity	Opportunity
Type	Type
SR#	Service Request
Status	Status

Display Name	Field Name
Repeating	Repeating
Frequency	Frequency
Repeating Expires	Repeating Expires
Owned By	Owned By
Comment	Comment

Activity Form Applet

The following lists the applet's details.

Applet Name Activity Form Applet
Business Component Action

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Description	Description
Type	Type
Comments	Comment
Meeting Location*	MeetingLocation
Private*	Private
Repeat Frequency*	Repeating Type
Repeat Until*	Repeating Expires
Activity #*	Id
Created By*	Created By Name
Created*	Created
Global Owner*	S-S Instance
Category*	Class
Include in Status Report*	Status Report Flag
Start	Planned
Duration	Duration Minutes
End	Planned Completion
Due	Due
Earliest Start*	No Sooner Than Date
Actual Start*	Started
Actual End*	Done
Done*	Done Flag
% Complete*	Percent Complete
Effort Remaining*	Est Work Time Remaining
Work Duration*	Call Duration
Lock Assignment*	Assignment Excluded
Audience*	Audience
Priority	Priority
Status	Status
Opportunity	Opportunity

Display Name	Field Name
Account	Account Name
Site*	Account Location
SR #*	SR Number
Change Request #*	Defect Num
Defective Tag*	Defective Tag
Resolution Code*	Resolution Code
Call ID*	Call Id
Asset #*	Asset Number
Serial #*	Serial Number
Orders*	Order Number
Source*	Campaign Name
Owner	Owned By
Employees	Owned By
Contacts	Contact Last Name
Display In	Display
Alarm*	Alarm
Alarm Lead*	Appt Alarm Time Min
Parent Activity*	Previous Activity Description
Project*	Project Name
Billable*	Billable Flag
Cost Estimate*	Associated Cost
Rate List*	Rate List
Price List*	Price List

NOTE: Fields marked by an asterisk (*) are not displayed by default in the user interface. XML queries to SWE can be set to include results from the hidden fields by using the ToggleLayout command. For an example, see “Retrieving Data from Hidden Fields” on page 41.

SMC Activity Contact List Applet

The following lists the applet's details.

Applet Name SMC Activity Contact List Applet
Business Component Contact

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Row ID	Id
Last Name	Last Name
First Name	First Name
Job Title	Job Title
Account	Account
Work Phone #	Work Phone #
Fax Phone #	Fax Phone #
Home Phone #	Home Phone #
Cellular Phone #	Cellular Phone #
Email Address	Email Address
Street Address	Street Address
City	City
State	State
Zip Code	Postal Code
Country	Country

Activity Attachment Applet

The following lists the applet's details.

Applet Name Activity Attachment Applet
Business Component Action Attachment

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Attachment Name	Activity FileName
Size (In Bytes)	Activity FileSize
Type	Activity FileExt
Modified	Activity FileDate
Update File	Activity FileAutoUpdFlg
Comments	Activity Comment

SMC Contact Screen

Information relating to your contacts is recorded and tracked in the views of the SMC Contact screen, as shown in Figure 17. Contacts are entities or individuals with whom the company does business or with whom it expects to do business in the future.

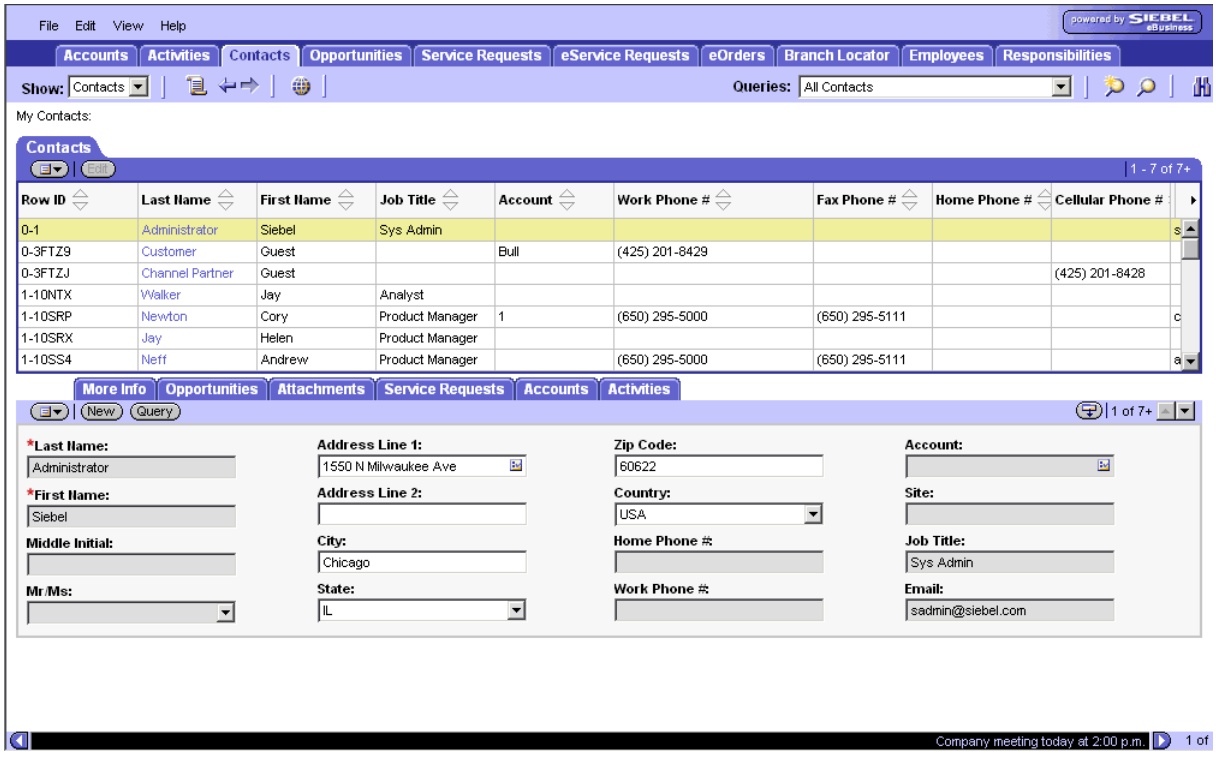


Figure 17. SMC Contact Screen

The following lists the details for the SMC Contact screen.

Display Name	Contacts
Business Object	Contact
Business Component	Contact

The following views are available from the SMC Contacts screen.

View Name	Viewbar Text	Sequence
SMC Contact View	Contacts	1
SMC Contact Detail View	More Info	2
SMC Contact Detail - Opportunities View	Opportunities	3
Contact Attachment View	Attachments	4
SMC Contact Detail - Service Request View	Service Requests	5
SMC Contact Detail - Accounts View	Accounts	6
SMC Contact Detail - Accounts View	Activities	7

SMC Contact View

The following lists the view details.

Screen	SMC Contact Screen
View	SMC Contact View
Business Object	Contact
Applet	SMC Contact List Applet, Contact Form Applet
Visibility	Organization

SMC Contact Detail View

The following lists the view details.

Screen	SMC Contact Screen
View	SMC Contact Detail View
Business Object	Contact
Applet	SMC Contact List Applet, Contact Form Applet
Visibility	Organization

SMC Contact Detail - Opportunities View

The following lists the view details.

Screen	SMC Contact Screen
View	SMC Contact Detail - Opportunities View
Business Object	Contact
Applet	SMC Oppty List Applet
Visibility	Organization

Contact Attachment View

The following table describes the view details.

Screen	SMC Contact Screen
View	Contact Attachment View
Business Object	Contact
Applet	Contact Attachment Applet
Visibility	Organization

SMC Contact Detail - Service Request View

The following lists the view details.

Screen	SMC Contact Screen
View	SMC Contact Detail - Service Request View
Business Object	Contact
Applet	SMC Service Request List Applet
Visibility	Organization

SMC Contact Detail - Accounts View

The following lists the view details.

Screen	SMC Contact Screen
View	SMC Contact Detail - Accounts View
Business Object	Contact
Applet	SMC Contact Account List Applet
Visibility	Account

SMC Contact Detail - Activities View

The following lists the view details.

Screen	SMC Contact Screen
View	SMC Contact Detail - Activities View
Business Object	Contact
Applet	SMC Activity List Applet
Visibility	Action

SMC Contact List Applet

The following lists the applet's details.

Applet Name	SMC Contact List Applet
Business Component	Contact

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Row ID	Id
Last Name	Last Name
First Name	First Name
Job Title	Job Title
Account	Account
Work Phone #	Work Phone #
Fax Phone #	Fax Phone #
Home Phone #	Home Phone #
Cellular Phone #	Cellular Phone #
Email Address	Email Address
Street Address	Street Address
City	City
State	State
ZIP Code	Postal Code
Country	Country

Contact Form Applet

The following lists the applet's details.

Applet Name	Contact Form Applet
Business Component	Contact

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Last Name	Last Name
First Name	First Name
Middle Initial	Middle Name
Mr/Mrs	M/M
Status*	Status
Households*	Households
Household Status*	Household Status
Manager Last Name*	Manager Last Name
Manager First Name*	Manager First Name
Contact Team*	Sales Rep
Comments*	Comment
Address Line 1	Street Address
Address Line 2	Street Address 2
City	City
State	State
Mail Stop*	Mail Stop
Contact Method*	Preferred Communications
Gender*	M/F
Organization*	Organization
Registration Source*	Registration Source App Name
Global Owner*	S-S Instance
Time Zone*	Time Zone Name - Translation
Account Status*	Account Status
Zip Code	Postal Code
Country	Country
Home Phone #	Home Phone #
Work Phone #	Work Phone #
Work Phone Extension*	Work Phone #
Work Fax #*	Fax Phone #

Display Name	Field Name
Mobile Phone #*	Cellular Phone #
Alternate Phone #*	Alternate Phone #
Alternate Email*	Alternate Email Address
Assistant Name*	Assistant
Assistant Phone #*	Assistant Phone #
Sync List*	PIM Sync Owner
Account	Account
Site	Account Location
Job Title	Job Title
Email	Email Address
Lock Assignment*	Assignment Excluded
Employee Flag*	Employee Flag
Send Email Updates*	Email SR Updates Flag
Never Email*	Suppress All Emails
Never Call*	Suppress All Calls
Never Mail*	Suppress All Mailings
Disable Cleaning*	Disable DataCleansing
Sync*	PIM Current User Sync Flag

NOTE: Fields marked by an asterisk (*) are those not displayed by default in the user interface. XML queries to SWE can be set to include results from the hidden fields by using the ToggleLayout command. For an example, see "Retrieving Data from Hidden Fields" on page 41.

SMC Oppty List Applet

The following lists the applet's details.

Applet Name SMC Oppty List Applet
Business Component Opportunity

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Name	Name
Account	Account
Primary	Primary Sales Rep Login
Revenue	Primary Revenue Amount
Sales Stage	Sales Stage
Close Date	Primary Revenue Close Date
New	Row Status
Priority Flag	Priority Flag
Status	Status
Site	Account Location
Committed	Primary Revenue Committed Flag
Expected Value	Expected Value
Revenue Class	Primary Revenue Class
Probability %	Primary Revenue Win Probability
Channel	Channel
Description	Description
Lead Quality	Quality
Reason	Reason Won Lost
Revenue Type	Primary Revenue Type
Sales Method	Sales Method
Source	Source
Lead Partner	Partner
Executive Priority	Executive Priority Flag
Executive Priority Date	Executive Priority Date

Contact Attachment Applet

The following lists the applet's details.

Applet Name SMC Contact Attachment Applet
Business Component Contact Attachment

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Attachment Name	FileName
Size (In Bytes)	FileSize
Type	FileExt
Modified	FileDate
Local	DockStatus
Download File	FileDockReqFlg
Update File	FileAutoUpdFlg
Comments	Comment

SMC Service Request List Applet

The following lists the applet's details.

Applet Name SMC Service Request List Applet
Business Component Service Request

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
New	Row Status
Row ID	Id
SR #	SR Number
Account	Account
Site	Account Location

Display Name	Field Name
Description	Description
Contact First Name	Contact First Name
Contact Last Name	Contact Last Name
Owner	Owner
Priority	Priority
Status	Status
Substatus	Sub-Status
Notes/Comments	Abstract
Phone #	Main Phone #
Product	Product
Area	Area
Subarea	Sub-Area
Committed	Commit Time

SMC Contact Account List Applet

The following lists the applet's details.

Applet Name SMC Contact Account List Applet
Business Component Contact

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Primary	SSA Primary Field
New	Row Status
Name	Name
Site	Location
Main Phone #	Main Phone Number
Territories	Territory
Industries	Industry
Status	Account Status

Display Name	Field Name
URL	Home Page
Alias	Alias
Current Volume	Current Volume
Potential Volume	Total Potential Volume
CSN	CSN
DUNS #	DUNS Number
Account Team	Sales Rep
Address Line 1	Street Address
Address Line 2	Street Address 2
City	City
State	State
Zip Code	Postal Code
Country	Country

SMC Activity List Applet

The following lists the applet's details.

Applet Name SMC Activity List Applet
Business Component Action

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Row ID	Id
New	Row Status
Description	Description
Planned Start	Planned
Planned End	Planned Completion
Priority	Priority
Contact Last Name	Contact Last Name

Display Name	Field Name
Contact First Name	Contact First Name
Account	Account Name
Account Site	Account Location
Opportunity	Opportunity
Type	Type
Status	Status
Repeating	Repeating
Frequency	Repeating Type
Repeating Expires	Repeating Expires
Owned By	Owned By
Comment	Comment

SMC Opportunity Screen

Information relating to opportunities is recorded and tracked in the views of the SMC Opportunity screen, as shown in Figure 18. An opportunity is defined as a potential revenue-generating event.

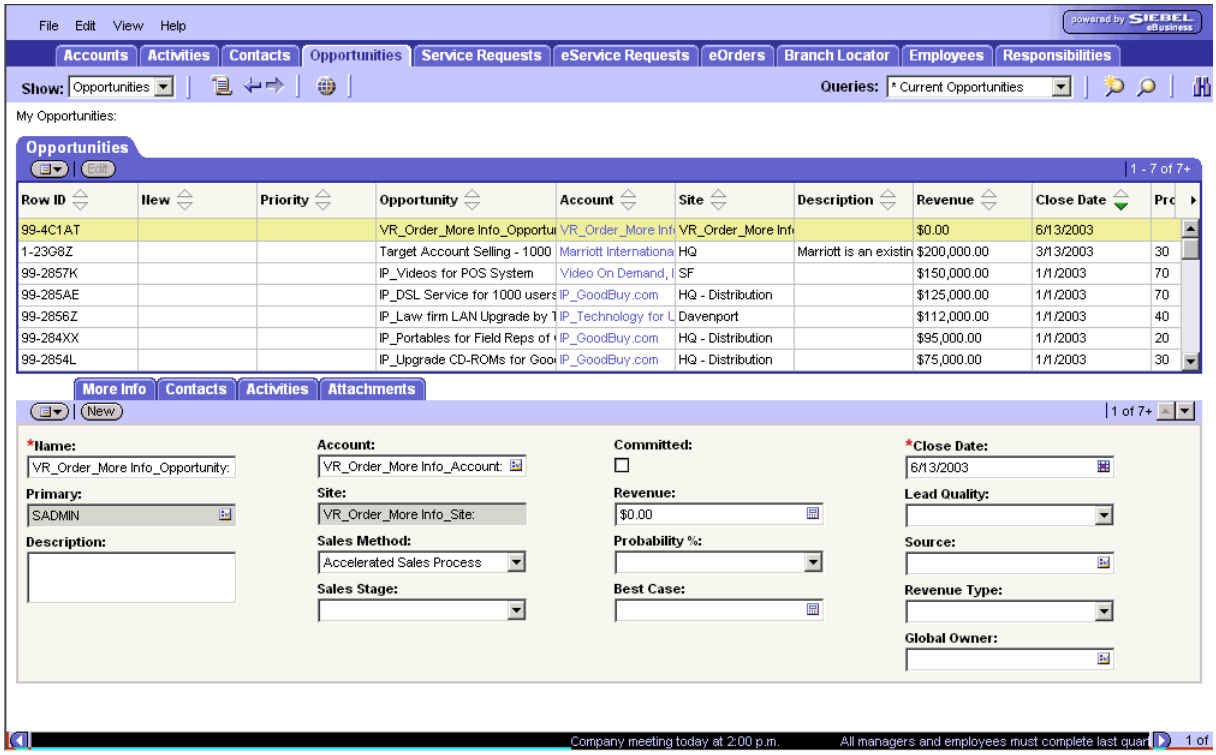


Figure 18. SMC Opportunities Screen

The following lists the details for the SMC Opportunity screen.

Display Name	Opportunities
Business Object	Opportunity
Business Component	Opportunity

The following views are available from the SMC Opportunity screen.

View Name	Viewbar Text	Sequence
SMC Opportunity View	Opportunities	1
SMC Opportunity Detail View	More Info	2
SMC Opportunity Detail - Contacts View	Contacts	3

View Name	Viewbar Text	Sequence
SMC Opportunity Detail - Activities View	Activities	4
Opportunity Attachment View	Attachments	5

SMC Opportunity View

The following lists the view details.

Screen	SMC Opportunity Screen
View	SMC Opportunity View
Business Object	Opportunity
Applet	SMC Opportunity List Applet, Opportunity Form Applet
Visibility	Sales Rep

SMC Opportunity Detail View

The following lists the view details.

Screen	SMC Opportunity Screen
View	SMC Opportunity Detail View
Business Object	Opportunity
Applet	SMC Opportunity List Applet, Opportunity Form Applet
Visibility	Sales Rep

SMC Opportunity Detail - Contacts View

The following lists the view details.

Screen	SMC Opportunity Screen
View	SMC Opportunity Detail - Contacts View
Business Object	Opportunity
Applet	SMC Opportunity Contact List Applet
Visibility	Sales Rep

SMC Opportunity Detail - Activities View

The following lists the view details.

Screen	SMC Opportunity Screen
View	SMC Opportunity Detail - Activities View
Business Object	Opportunity
Applet	SMC Opportunity Activity List Applet
Visibility	Sales Rep

Opportunity Attachment View

The following lists the view details.

Screen	SMC Opportunity Screen
View	Opportunity Attachment View
Business Object	Opportunity
Applet	Opportunity Attachment View
Visibility	Sales Rep

SMC Opportunity List Applet

The following lists the applet's details.

Applet Name	SMC Opportunity List Applet
Business Component	Opportunity

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Row ID	Id
New	Row Status
Priority	Priority Flag
Opportunity	Name
Account	Account
Site	Account Site
Description	Description

Display Name	Field Name
Revenue	Primary Revenue Amount
Close Date	Primary Revenue Close Date
Probability %	Primary Revenue Win Probability
Sales Method	Sales Method
Sales Stage	Sales Stage
Last Name	Contact First Name
First Name	Contact Last Name
Committed	Committed
Primary	Primary Sales Rep Login

Opportunity Form Applet

The following lists the applet's details.

Applet Name Opportunity Form Applet
Business Component Opportunity

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
New	Row Status
Priority	Priority Flag
Name	Name
Account	Account
Site	Account Location
Description	Description
Revenue	Primary Revenue Amount
Close Date	Primary Revenue Close Date
Probability %	Primary Revenue Win Probability
Sales Method	Sales Method
Sales Stage	Sales Stage
Contact Last Name	Key Contact Last Name

Display Name	Field Name
Contact First Name	Key Contact First Name
Committed	Committed
Primary	Primary Sales Rep Login

SMC Opportunity Contact List Applet

The following lists the applet's details.

Applet Name SMC Opportunity Contact List Applet
Business Component Contact

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Last Name	Last Name
First Name	First Name
Mr/Mrs	M/M
Work Phone #	Work Phone #
Job Title	Job Title
Email	Email Address
Account	Account
Site	Account Location
Role	Role
Employee Flag	Employee Flag
Middle Initial	Middle Name
Contacted	Contacted
Work Fax #	Fax Phone #
Mobile Phone #	Cellular Phone #
Comment	Opportunity Contact Comment
Full Name	Full Name
Nickname	Alias

SMC Opportunity Activity List Applet

The following lists the applet's details.

Applet Name SMC Opportunity Activity List Applet
Business Component Action

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Row ID	Id
New	Row Status
Description	Description
Planned Start	Planned
Planned End	Planned Completion
Priority	Priority
Contact Last Name	Contact Last Name
Contact First Name	Contact First Name
Account	Account Name
Account Site	Account Location
Opportunity	Opportunity
Type	Type
SR #	SR Number
Status	Status
Repeating	Repeating
Frequency	Repeating Type
Repeating Expires	Repeating Expires
Owned By	Owned By
Comment	Comment

Opportunity Attachment Applet

The following lists the applet's details.

Applet Name	Opportunity Attachment Applet
Business Component	Opportunity Attachment

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Attachment Name	Oppty FileName
Size (In Bytes)	Oppty FileSize
Type	Oppty File Ext
Modified	Oppty File Date
Local	Oppty DockStatus
Download File	Oppty FileDocReqFlg
Update File	Oppty FileAutoUpdFlg
Comments	Oppty Comment

SMC Service Request Screen

Information relating to service requests is recorded and tracked in the views of the SMC Opportunities screen, as shown in Figure 19. A service request is a customer request for information about or assistance with products or services bought from a company.

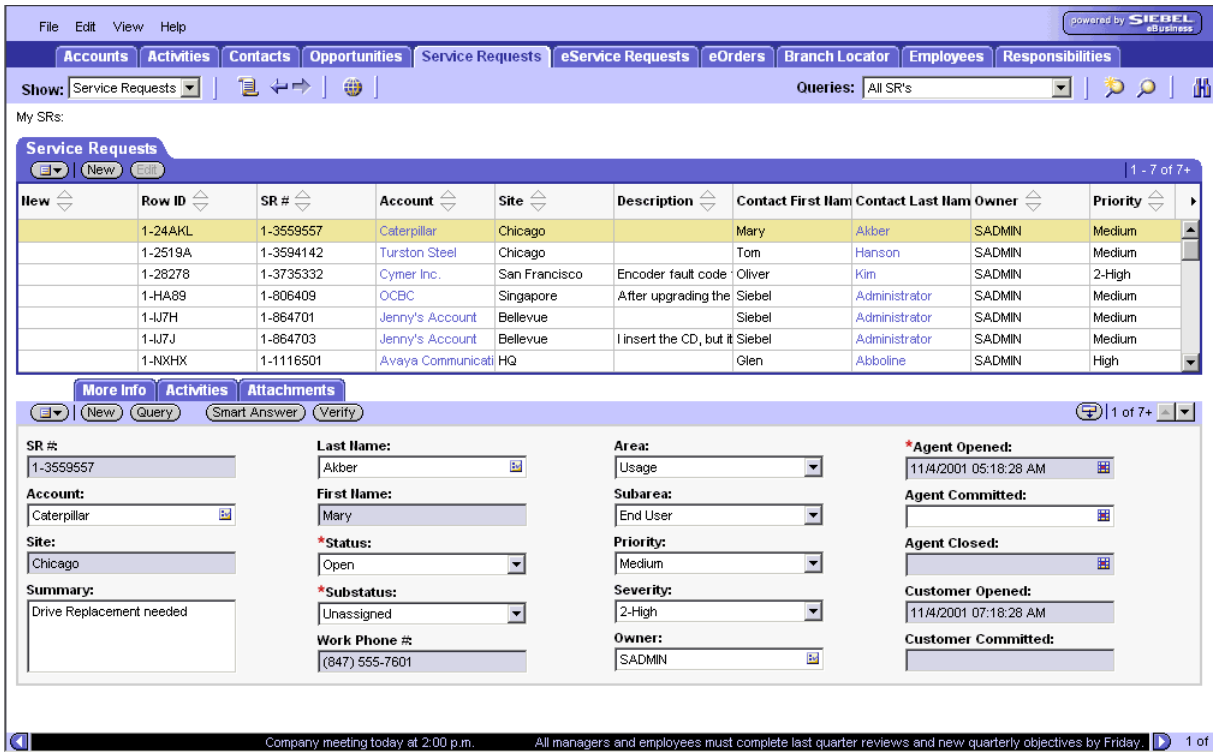


Figure 19. SMC Service Request Screen

The following lists the details for the SMC Service Request screen.

Display Name Service Requests
Business Object Service Request
Business Component Service Request

The following views are available from the SMC Account screen.

View Name	Viewbar Text	Sequence
SMC Service Request View	Service Requests	1
SMC Service Request Detail View	More Info	2
Service Request Detail View	Activities	3
Service Request Detail View with Attachments	Attachments	4

SMC Service Request View

The following lists the view details.

Screen	SMC Service Request Screen
View	SMC Service Request View
Business Object	Service Request
Applet	SMC Service Request List Applet, Service Request Detail Applet
Visibility	Service Request

SMC Service Request Detail View

The following lists the view details.

Screen	SMC Service Request Screen
View	SMC Service Request Detail View
Business Object	Service Request
Applet	SMC Service Request List Applet, Service Request Detail Applet
Visibility	Service Request

Service Request Detail View

The following lists the view details.

Screen	SMC Service Request Screen
View	Service Request Detail View
Business Object	Service Request
Applet	Service Request Activity List Applet
Visibility	Service Request

Service Request Detail View with Attachments

The following lists the view details.

Screen	SMC Service Request Screen
View	Service Request Detail View with Attachments
Business Object	Service Request
Applet	Service Request Attachment List Applet
Visibility	Service Request

SMC Service Request List Applet

The following lists the applet's details.

Applet Name SMC Service Request List Applet
Business Component Service Request

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
New	Row Status
Row ID	Id
SR #	SR Number
Account	Account
Site	Account Location
Description	Description
Contact First Name	Contact First Name
Contact Last Name	Contact Last Name
Owner	Owner
Priority	Priority
Status	Status
Substatus	Sub-Status
Notes/Comments	Abstract
Phone #	Main Phone Number
Product	Product
Area	Area
Subarea	Sub-Area
Committed	Commit Time

Service Request Detail Applet

The following lists the applet's details.

Applet Name SMC Service Request Detail Applet
Business Component Service Request

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
SR #	SR Number
Account	Account
Site	Account Location
Summary	Abstract
Description*	Description
Last Name	Contact Last Name
First Name	Contact First Name
Status	Status
Substatus	Sub-Status
Work Phone #	Contact Business Phone
Contact Account*	Contact Account
Customer Reference ID*	Customer Ref Number
Source*	Source
Entitlement*	Entitlement Name
Organization*	Organization
Area	Area
Subarea	Sub-Area
Priority	Priority
Severity	Severity
Owner	Owner
Group*	Owner Group
Reproducible*	Reproduce
Billable*	Billable Flag
Rate List*	Rate List
Price List*	Price List
Agent Opened	Created
Agent Committed	Commit Time
Agent Closed	Closed Date
Customer Opened	Contact Created

Display Name	Field Name
Customer Committed	Contact Commit Time
Customer Closed*	Contact Closed Date
Product*	Product
Part #*	Product Part Number
Asset #*	Asset Number
Profile*	Profile Product Name

NOTE: Fields marked by an asterisk (*) are those not displayed by default in the user interface. XML queries to SWE can be set to include results from the hidden fields by using the ToggleLayout command. For an example, see "Retrieving Data from Hidden Fields" on page 41.

Service Request Activity List Applet

The following lists the applet's details.

Applet Name Service Request Activity List Applet
Business Component Action

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
New	Row Status
Description	Description
Type	Type
Start	Planned
Due	Due
Status	Status
Priority	Priority
Owner	Primary Owned By
Account	Account Name
Display In	Display
Site	Account Location
Opportunity	Opportunity

Display Name	Field Name
Activity #	Id
Private	Private
Parent Activity	Previous Activity Description
Comments	Comment
Meeting Location	Meeting Location
End	Planned Completion
Repeat Frequency	Repeating Type
Repeat Until	Repeating Expires
Alarm	Alarm
Attachments	Attachment Flag
Created	Created
Created By	Created By Name
Actual End	Done
Done	Done Flag
Employees	Owned By
Contacts	Contact Last Name
Service Region	Service Region

Service Request Attachment List Applet

The following lists the applet's details.

Applet Name Service Request Attachment List Applet
Business Component Service Request Attachment

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Attachment Name	Activity FileName
Size (In Bytes)	Activity FileSize
Type	Activity FileExt

Display Name	Field Name
Modified	Activity FileDate
Local	ActivityDockStatus
Download File	ActivityFileDockReqFlg
Update File	Activity FileAutoUpdFlg
Comments	Activity Comments
Created	Activity Created

SMC eService Request Screen

Information relating to eService requests is recorded and tracked in the views of the SMC eService Request screen, as shown in Figure 20. An eService request is a customer view of a service request.

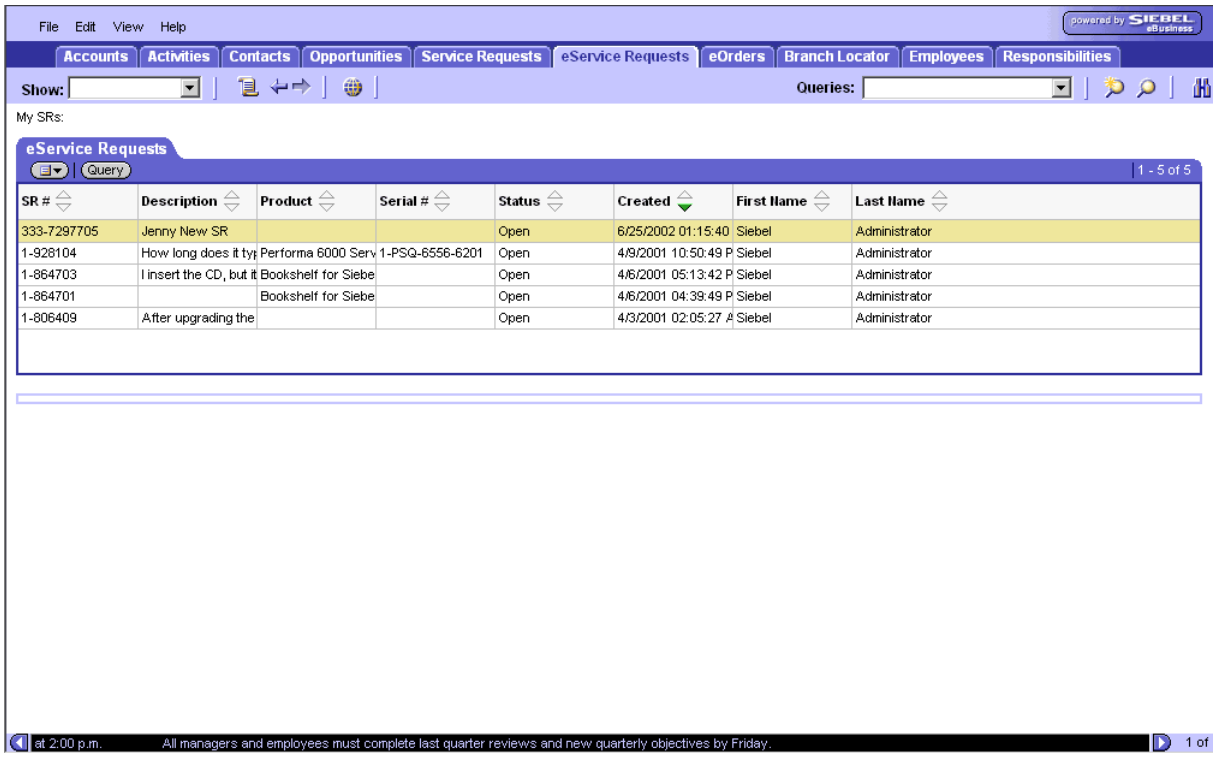


Figure 20. SMC eService Request Screen

NOTE: This is a read-only screen.

The following lists the details for the SMC eService Request screen.

Display Name	eService Requests
Business Object	Service Request (eService)
Business Component	Service Request (eService)

The following view is available from the SMC Account screen.

View Name	Viewbar Text	Sequence
SMC eService Request View	eService Requests	1

SMC eService Request View

The following lists the view details.

Screen	SMC Service Request Screen
View	SMC eService Request View
Business Object	Service Request (eService)
Applet	SMC eService Request List Applet
Visibility	Personal

SMC eService Request List Applet

The following lists the applet's details.

Applet Name	SMC eService Request List Applet
Business Component	Service Request (eService)

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
SR #	Id
Description	Description
Product	Product
Serial #	Serial Number
Status	Status
Created	Created

Display Name	Field Name
First Name	Contact First Name
Last Name	Contact Last Name

SMC eService Order Screen

Information relating to orders is tracked in the views of the SMC Orders screen, as shown in Figure 21. An order is an order for products and services, including replacement or repair of parts.

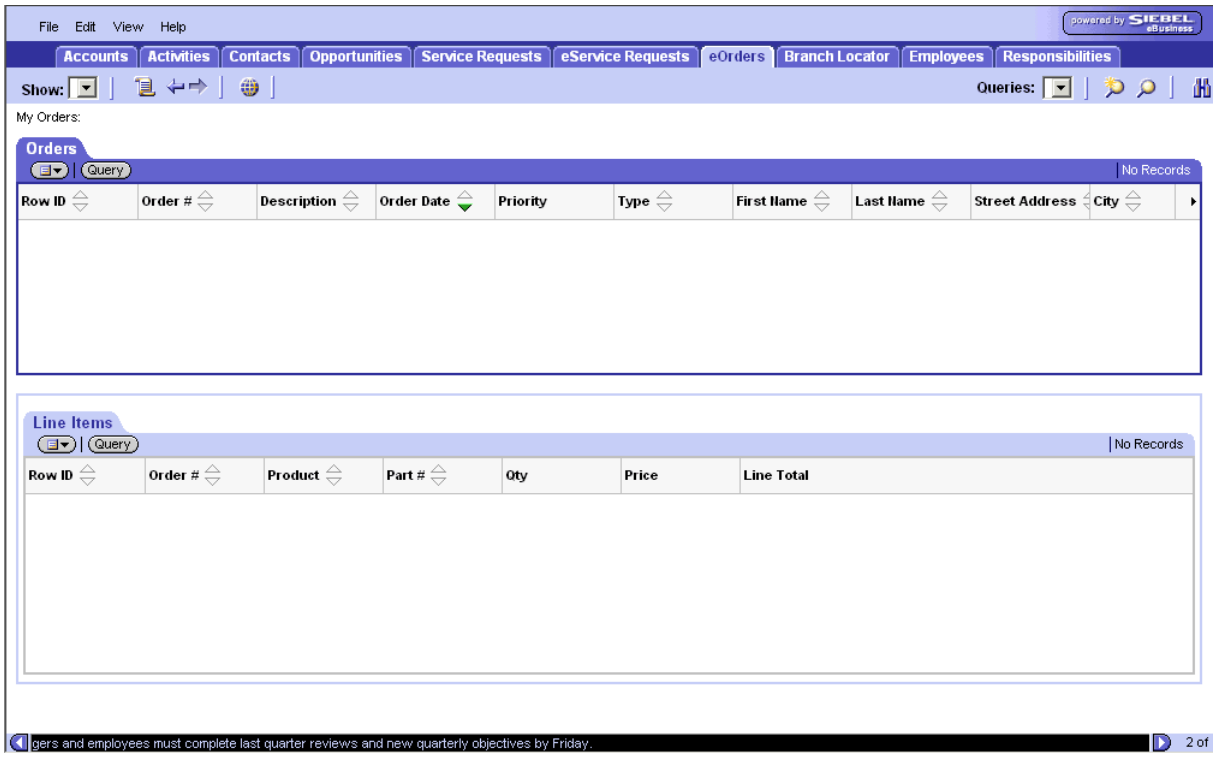


Figure 21. SMC Orders Screen

NOTE: This is a read-only screen.

The following lists the details for the SMC Orders screen.

Display Name eOrders
Business Object Order Entry (eService)
Business Component Order Entry - Orders

The following views are available from the SMC Orders screen.

View Name	Viewbar Text	Sequence
SMC eService Order View	Orders	1

SMC eService Order View

The following lists the view details.

Screen SMC eService Order Screen
View SMC eService Order View
Business Object Order Entry (eService)
Applet SMC eService Orders List Applet
Visibility Personal

SMC eService Orders List Applet

The following lists the applet's details.

Applet Name SMC eService Orders List Applet
Business Component Order Entry - Orders

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Row ID	Id
Order #	Order Number
Description	Description
Order Date	Order Date
Priority	Priority Status
Type	Order Type
First Name	Contact First Name
Last Name	Contact Last Name
Street Address	ShipTo - Address
City	ShipTo - City

Display Name	Field Name
State	ShipTo - State
Zip Code	ShipTo - Zip
Country	ShipTo - Country
Carrier	Carrier Type
Credit Card Name	Credit Card Name

SMC eOrder Line Items List Applet

The following lists the applet's details.

Applet Name SMC eOrder Line Items List Applet
Business Component Order Entry - Line Items

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Row ID	Id
Order #	Order
Product	Item
Part #	Part #
Qty	Quantity
Price	Item Price
Line Total	Line Total

SMC Branch Locator Screen

Information relating to branch locations is tracked in the views of the SMC Branch Locator screen, as shown in Figure 22. Branch Locator allows customers to identify the stores and branches of an organization that are closest to a specified location, such as their home or work.

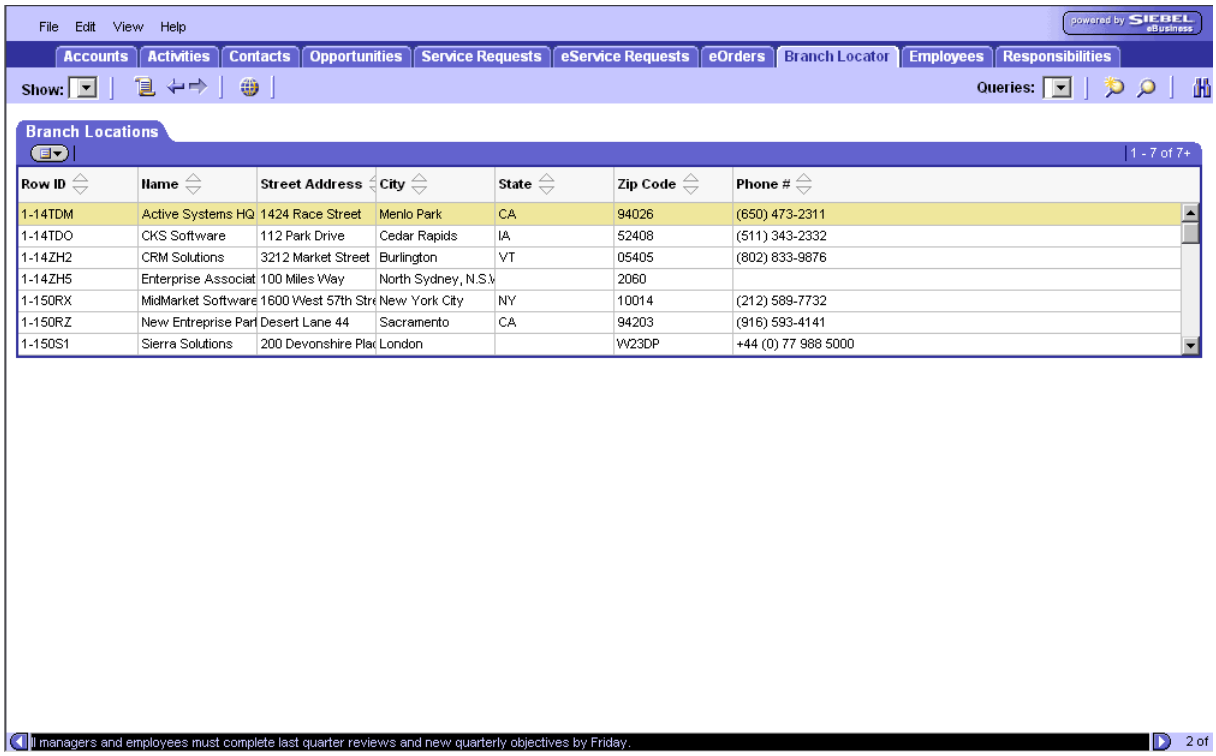


Figure 22. SMC Branch Locator Screen

NOTE: This is a read-only screen.

The following lists the details for the SMC Branch Locator screen.

Display Name	Branch Locator
Business Object	Service Locator
Business Component	Service Locator

The following view is available from the SMC Branch Locator screen.

View Name	Viewbar Text	Sequence
SMC Branch Locator View	Branch Locator	1

SMC Branch Locator View

The following lists the view details.

Screen	SMC Branch Locator Screen
View	SMC Branch Locator View
Business Object	Service Locator
Applet	SMC Branch Locator
Visibility	All

SMC Branch Locator

The following lists the applet's details.

Applet Name	SMC Branch Locator
Business Component	Service Locator

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Row ID	Id
Name	Service Name
Street Address	Street Address
City	City
State	State
Zip Code	Zip Code
Phone #	Phone Number

SMC Employee Screen

Information relating to employees is tracked in the views of the SMC Employee screen, as shown in Figure 23. An employee represents someone who works for your company, including sales representatives and service technicians.

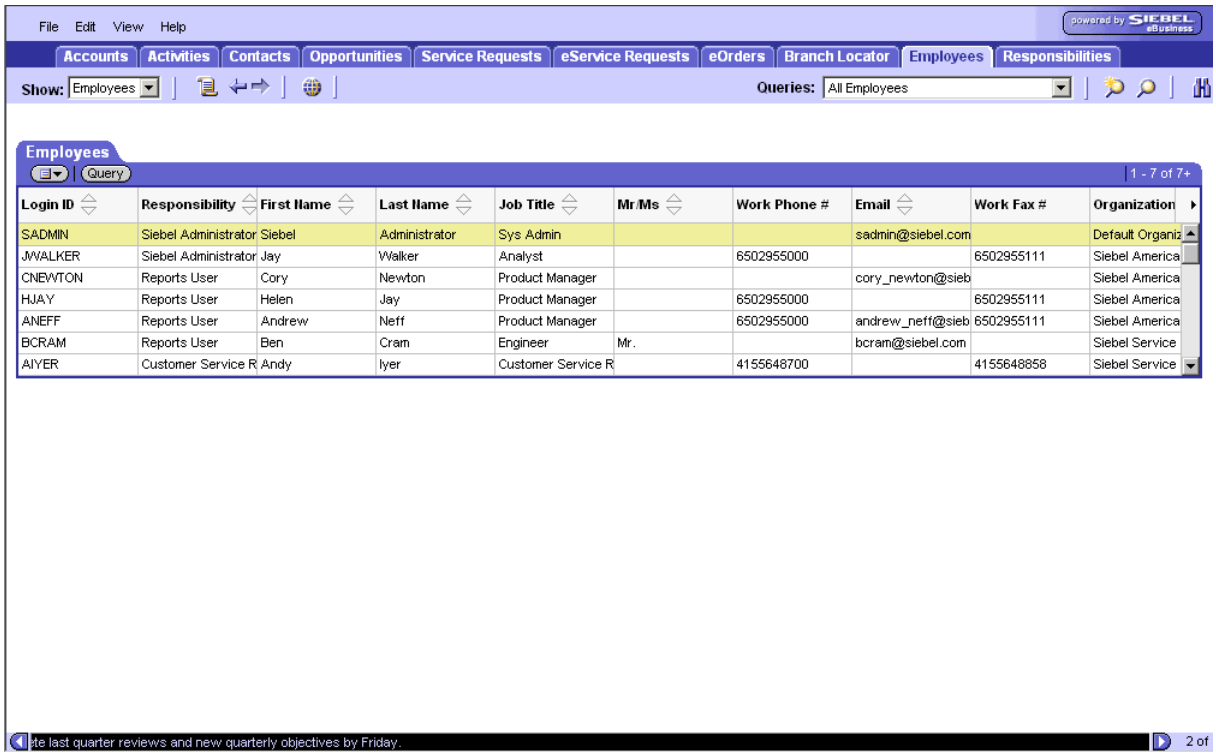


Figure 23. SMC Employee Screen

NOTE: This is a read-only screen.

The following lists the details for the SMC Employees screen.

Display Name	Employees
Business Object	Employee
Business Component	Employee

The following view is available from the SMC Employees screen.

View Name	Viewbar Text	Sequence
SMC Employee View	Employees	1

SMC Employee View

The following lists the view details.

Screen	SMC Employee Screen
View	SMC Employee View
Business Object	Employee
Applet	SMC Employee ReadOnly List Applet
Visibility	Organization

SMC Employee ReadOnly List Applet

The following lists the applet's details.

Applet Name	SMC Employee ReadOnly List Applet
Business Component	Employee

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Login ID	Login Name
Responsibility	Responsibility
First Name	First Name
Last Name	Last Name
Job Title	Job Title
Mr/Mrs	Personal Title
Work Phone #	Phone Number
Email	Email Addr
Work Fax #	Fax Number
Organization	Organization
Division	Division
Cell Phone #	Cell Phone #
Street Address	Street Address
City	City
State	State

Display Name	Field Name
Zip Code	Postal Code
Home Phone #	Home Phone #
Country	Country
Alias	Alias

SMC Responsibility Screen

Information relating to responsibilities, users and views is tracked in the views of the SMC Responsibility screen, as shown in Figure 24. An employee represents someone who works for your company, including sales representatives and service technicians. A user is someone who has been assigned by the administrator to access the views associated with a responsibility.

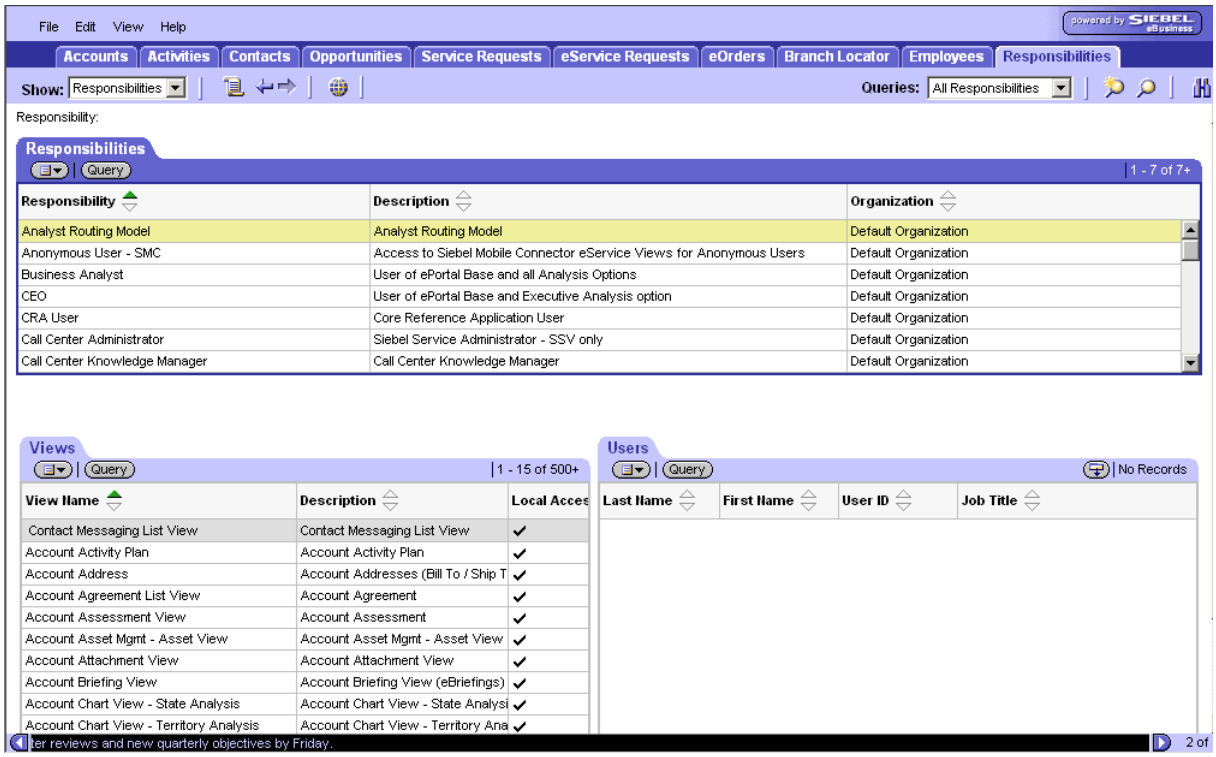


Figure 24. SMC Responsibility Screen

NOTE: This is a read-only screen.

The following lists the details for the SMC Responsibilities screen.

Display Name Responsibilities
Business Object Responsibility

The following views or applets are available from the SMC Account screen.

View Name	Viewbar Text	Sequence
SMC Responsibility View	Responsibilities	1

SMC Responsibility View

The following lists the view details.

Screen SMC Responsibility Screen
View SMC Responsibility View
Business Object Responsibility
Applet SMC Responsibility List Administration Applet, SMC View List Administration Applet
Visibility All

SMC Responsibility List Administration Applet

The following lists the applet's details.

Applet Name SMC Responsibility List Administration Applet
Business Component Responsibility

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Responsibility	Name
Description	Description
Organization	Organization

SMC View List Administration Applet

The following lists the applet's details.

Applet Name	SMC View List Administration Applet
Business Component	Feature Access

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
View Name	Name
Description	Description
Local Access	Local Access

SMC User List Administration Applet

The following lists the applet's details.

Applet Name	SMC User List Administration Applet
Business Component	User

List Columns and Controls

The following table lists the available fields and their display names.

Display Name	Field Name
Last Name	Last Name
First Name	First Name
User ID	Login Name
Job Title	Job Title
Alias	Alias
MI	Middle Name
Short Name	Nick Name
Mr/Mrs	Personal Title
Email	EMail Addr
Work Phone #	Phone #

Display Name	Field Name
Work Fax #	Fax #
Responsibility	Responsibility
Home Phone #	Home Phone #
Share Home Phone	Share Home Phone Flag
Share Address	Share Address Flag

B Sample XML and XSL Output

This appendix contains sample XML and XSL output from the Siebel Mobile Connector, including a screen representation, an alert message, and a style sheet.

Sample XML Screen Representation

The following XML code contains a complete representation of the SMC Employees Screen (data only view):

```
<?xml version="1.0" ?>

<APPLICATION NAME="Siebel Mobile Connector">

   <USER_AGENT MARKUP="HTML" />

   <SCREEN CAPTION="Employees" ACTIVE="TRUE" NAME="SMC Employee Screen">

      <VIEW ACTIVE="TRUE" TITLE="Employees" NAME="SMC Employee View">

         <APPLET MODE="Base" NO_INSERT="FALSE" ACTIVE="TRUE" CLASS="CSSFrameList" TITLE="Employees" ID="1"
NO_MERGE="FALSE" NO_DELETE="FALSE" NO_UPDATE="FALSE" NO_EXEC_QUERY="FALSE" NAME="SMC Employee ReadOnly
List Applet">

            <LIST>

               <RS_HEADER>

                  <COLUMN HTML_TYPE="Text" DISPLAY_NAME="Last Name" SCALE="0" TEXT_LENGTH="50"
DATATYPE="text" HIDDEN="FALSE" NUMBER_BASED="FALSE" LIST_EDITABLE="Y" ID="501" READ_ONLY="FALSE"
REQUIRED="TRUE" TEXT_BASED="TRUE" TYPE="Text" CALCULATED="FALSE" TOTAL_REQUIRED="FALSE" FIELD="Last Name"
NAME="Last Name" />

                  <COLUMN HTML_TYPE="Text" DISPLAY_NAME="First Name" SCALE="0" TEXT_LENGTH="50"
DATATYPE="text" HIDDEN="FALSE" NUMBER_BASED="FALSE" LIST_EDITABLE="Y" ID="502" READ_ONLY="FALSE"
REQUIRED="TRUE" TEXT_BASED="TRUE" TYPE="Text" CALCULATED="FALSE" TOTAL_REQUIRED="FALSE" FIELD="First Name"
NAME="First Name" />

                  <COLUMN HTML_TYPE="Text" DISPLAY_NAME="Login ID" SCALE="0" TEXT_LENGTH="50"
DATATYPE="text" HIDDEN="FALSE" NUMBER_BASED="FALSE" LIST_EDITABLE="Y" ID="503" READ_ONLY="FALSE"
REQUIRED="TRUE" TEXT_BASED="TRUE" TYPE="Text" CALCULATED="FALSE" TOTAL_REQUIRED="FALSE" FIELD="Login Name"
NAME="Login Name" />

                  <COLUMN HTML_TYPE="Text" DISPLAY_NAME="Responsibility" SCALE="0" TEXT_LENGTH="50"
DATATYPE="text" HIDDEN="FALSE" NUMBER_BASED="FALSE" LIST_EDITABLE="Y" ID="504" READ_ONLY="FALSE"
REQUIRED="FALSE" TEXT_BASED="TRUE" TYPE="Text" CALCULATED="FALSE" TOTAL_REQUIRED="FALSE"
FIELD="Responsibility" NAME="Responsibility" />

                  <COLUMN HTML_TYPE="Text" DISPLAY_NAME="Job Title" SCALE="0" TEXT_LENGTH="75"
DATATYPE="text" HIDDEN="FALSE" NUMBER_BASED="FALSE" LIST_EDITABLE="Y" ID="505" READ_ONLY="FALSE"
REQUIRED="FALSE" TEXT_BASED="TRUE" TYPE="Text" CALCULATED="FALSE" TOTAL_REQUIRED="FALSE" FIELD="Job Title"
NAME="Job Title" />

                  <COLUMN HTML_TYPE="Text" DISPLAY_NAME="Email" SCALE="0" TEXT_LENGTH="100"
DATATYPE="text" HIDDEN="FALSE" NUMBER_BASED="FALSE" LIST_EDITABLE="Y" ID="506" READ_ONLY="FALSE"
REQUIRED="FALSE" TEXT_BASED="TRUE" TYPE="Text" CALCULATED="FALSE" TOTAL_REQUIRED="FALSE" FIELD="EMail
Addr" NAME="EMail Addr" />
```

```
                <COLUMN HTML_TYPE="Text" DISPLAY_NAME="Work Phone #" SCALE="0" TEXT_LENGTH="40"
DATATYPE="phone" HIDDEN="FALSE" NUMBER_BASED="FALSE" LIST_EDITABLE="Y" ID="507" READ_ONLY="FALSE"
REQUIRED="FALSE" TEXT_BASED="TRUE" TYPE="Text" CALCULATED="FALSE" TOTAL_REQUIRED="FALSE" FIELD="Phone
Number" NAME="Phone Number" />

                <COLUMN HTML_TYPE="Text" DISPLAY_NAME="Work Fax #" SCALE="0" TEXT_LENGTH="40"
DATATYPE="phone" HIDDEN="FALSE" NUMBER_BASED="FALSE" LIST_EDITABLE="Y" ID="508" READ_ONLY="FALSE"
REQUIRED="FALSE" TEXT_BASED="TRUE" TYPE="Text" CALCULATED="FALSE" TOTAL_REQUIRED="FALSE" FIELD="Fax
Number" NAME="Fax Number" />

                <COLUMN HTML_TYPE="Text" DISPLAY_NAME="Organization" SCALE="0" TEXT_LENGTH="100"
DATATYPE="text" HIDDEN="FALSE" NUMBER_BASED="FALSE" LIST_EDITABLE="Y" ID="509" READ_ONLY="FALSE"
REQUIRED="FALSE" TEXT_BASED="TRUE" TYPE="Text" CALCULATED="FALSE" TOTAL_REQUIRED="FALSE"
FIELD="Organization" NAME="Organization" />

                <COLUMN HTML_TYPE="Text" DISPLAY_NAME="Division" SCALE="0" TEXT_LENGTH="100"
DATATYPE="text" HIDDEN="FALSE" NUMBER_BASED="FALSE" LIST_EDITABLE="Y" ID="510" READ_ONLY="FALSE"
REQUIRED="FALSE" TEXT_BASED="TRUE" TYPE="Text" CALCULATED="FALSE" TOTAL_REQUIRED="FALSE" FIELD="Division"
NAME="Division" />

                <COLUMN HTML_TYPE="Text" DISPLAY_NAME="Mobile Phone #" LIST_EDITABLE="Y" ID="511"
READ_ONLY="FALSE" TYPE="Text" TOTAL_REQUIRED="FALSE" FIELD="Cellular Phone #" NAME="CellPhoneNo" />

                <COLUMN HTML_TYPE="Text" DISPLAY_NAME="Street Address" LIST_EDITABLE="Y" ID="512"
READ_ONLY="FALSE" TYPE="Text" TOTAL_REQUIRED="FALSE" FIELD="Street Address" NAME="StreetAddr" />

                <COLUMN HTML_TYPE="Text" DISPLAY_NAME="City" SCALE="0" TEXT_LENGTH="50" DATATYPE="text"
HIDDEN="FALSE" NUMBER_BASED="FALSE" LIST_EDITABLE="Y" ID="513" READ_ONLY="FALSE" REQUIRED="FALSE"
TEXT_BASED="TRUE" TYPE="Text" CALCULATED="FALSE" TOTAL_REQUIRED="FALSE" FIELD="City" NAME="City" />

                <COLUMN HTML_TYPE="Text" DISPLAY_NAME="State" SCALE="0" TEXT_LENGTH="10" DATATYPE="text"
HIDDEN="FALSE" NUMBER_BASED="FALSE" LIST_EDITABLE="Y" ID="514" READ_ONLY="FALSE" REQUIRED="FALSE"
TEXT_BASED="TRUE" TYPE="Text" CALCULATED="FALSE" TOTAL_REQUIRED="FALSE" FIELD="State" NAME="State" />

                <COLUMN HTML_TYPE="Text" DISPLAY_NAME="Zip Code" LIST_EDITABLE="Y" ID="515"
READ_ONLY="FALSE" TYPE="Text" TOTAL_REQUIRED="FALSE" FIELD="Postal Code" NAME="Zip" />

                <COLUMN HTML_TYPE="Text" DISPLAY_NAME="Country" SCALE="0" TEXT_LENGTH="30"
DATATYPE="text" HIDDEN="FALSE" NUMBER_BASED="FALSE" LIST_EDITABLE="Y" ID="516" READ_ONLY="FALSE"
REQUIRED="FALSE" TEXT_BASED="TRUE" TYPE="Text" CALCULATED="FALSE" TOTAL_REQUIRED="FALSE" FIELD="Country"
NAME="Country" />

                <COLUMN HTML_TYPE="Text" DISPLAY_NAME="Home Phone #" LIST_EDITABLE="Y" ID="517"
READ_ONLY="FALSE" TYPE="Text" TOTAL_REQUIRED="FALSE" FIELD="Home Phone #" NAME="Home Phone" />

                <COLUMN HTML_TYPE="Text" DISPLAY_NAME="Alias" SCALE="0" TEXT_LENGTH="50" DATATYPE="text"
HIDDEN="FALSE" NUMBER_BASED="FALSE" LIST_EDITABLE="Y" ID="518" READ_ONLY="FALSE" REQUIRED="FALSE"
TEXT_BASED="TRUE" TYPE="Text" CALCULATED="FALSE" TOTAL_REQUIRED="FALSE" FIELD="Alias" NAME="Alias" />

                <COLUMN HTML_TYPE="Text" DISPLAY_NAME="Mr/Ms" LIST_EDITABLE="Y" ID="519"
READ_ONLY="FALSE" TYPE="Text" TOTAL_REQUIRED="FALSE" FIELD="Personal Title" NAME="Mr/Mrs" />

              </RS_HEADER>

              <RS_DATA>

                 <ROW ROWID="0-1" SELECTED="TRUE">

                 <FIELD VARIABLE="Last Name" NAME="Last Name">Administrator</FIELD>

                 <FIELD VARIABLE="First Name" NAME="First Name">Siebel</FIELD>

                 <FIELD VARIABLE="Login Name" NAME="Login Name">SADMIN</FIELD>

                 <FIELD VARIABLE="Responsibility" NAME="Responsibility">Siebel Administrator</FIELD>

                 <FIELD VARIABLE="Job Title" NAME="Job Title">Sys Admin</FIELD>

                 <FIELD VARIABLE="EMail Addr" NAME="EMail Addr">sadmin@siebel.com</FIELD>

                 <FIELD VARIABLE="Phone Number" NAME="Phone Number" />
```

```
<FIELD VARIABLE="Fax Number" NAME="Fax Number" />
<FIELD VARIABLE="Organization" NAME="Organization">ŠD_Default Organization</FIELD>
<FIELD VARIABLE="Division" NAME="Division">ŠD_Siebel Administration</FIELD>
<FIELD VARIABLE="CellPhoneNo" NAME="Cellular Phone #" />
<FIELD VARIABLE="StreetAddr" NAME="Street Address" />
<FIELD VARIABLE="City" NAME="City" />
<FIELD VARIABLE="State" NAME="State" />
<FIELD VARIABLE="Zip" NAME="Postal Code" />
<FIELD VARIABLE="Country" NAME="Country" />
<FIELD VARIABLE="Home Phone" NAME="Home Phone #" />
<FIELD VARIABLE="Alias" NAME="Alias" />
<FIELD VARIABLE="Mr/Mrs" NAME="Personal Title" />
</ROW>
<ROW ROWID="0-3FTZ9">
<FIELD VARIABLE="Last Name" NAME="Last Name">Customer</FIELD>
<FIELD VARIABLE="First Name" NAME="First Name">Guest</FIELD>
<FIELD VARIABLE="Login Name" NAME="Login Name">GUESTCST</FIELD>
<FIELD VARIABLE="Responsibility" NAME="Responsibility">Web Anonymous User</FIELD>
<FIELD VARIABLE="Job Title" NAME="Job Title" />
<FIELD VARIABLE="EMail Addr" NAME="EMail Addr" />
<FIELD VARIABLE="Phone Number" NAME="Phone Number" />
<FIELD VARIABLE="Fax Number" NAME="Fax Number" />
<FIELD VARIABLE="Organization" NAME="Organization" />
<FIELD VARIABLE="Division" NAME="Division" />
<FIELD VARIABLE="CellPhoneNo" NAME="Cellular Phone #" />
<FIELD VARIABLE="StreetAddr" NAME="Street Address" />
<FIELD VARIABLE="City" NAME="City" />
<FIELD VARIABLE="State" NAME="State" />
<FIELD VARIABLE="Zip" NAME="Postal Code" />
<FIELD VARIABLE="Country" NAME="Country" />
<FIELD VARIABLE="Home Phone" NAME="Home Phone #" />
<FIELD VARIABLE="Alias" NAME="Alias" />
<FIELD VARIABLE="Mr/Mrs" NAME="Personal Title" />
</ROW>
<ROW ROWID="0-3FTZJ">
<FIELD VARIABLE="Last Name" NAME="Last Name">Channel Partner</FIELD>
```

```
                    <FIELD VARIABLE="First Name" NAME="First Name">Guest</FIELD>
                    <FIELD VARIABLE="Login Name" NAME="Login Name">GUESTCP</FIELD>
                    <FIELD VARIABLE="Responsibility" NAME="Responsibility">Unregistered Partner Agent</
FIELD>
                    <FIELD VARIABLE="Job Title" NAME="Job Title" />
                    <FIELD VARIABLE="EMail Addr" NAME="EMail Addr" />
                    <FIELD VARIABLE="Phone Number" NAME="Phone Number" />
                    <FIELD VARIABLE="Fax Number" NAME="Fax Number" />
                    <FIELD VARIABLE="Organization" NAME="Organization" />
                    <FIELD VARIABLE="Division" NAME="Division" />
                    <FIELD VARIABLE="CellPhoneNo" NAME="Cellular Phone #" />
                    <FIELD VARIABLE="StreetAddr" NAME="Street Address" />
                    <FIELD VARIABLE="City" NAME="City" />
                    <FIELD VARIABLE="State" NAME="State" />
                    <FIELD VARIABLE="Zip" NAME="Postal Code" />
                    <FIELD VARIABLE="Country" NAME="Country" />
                    <FIELD VARIABLE="Home Phone" NAME="Home Phone #" />
                    <FIELD VARIABLE="Alias" NAME="Alias" />
                    <FIELD VARIABLE="Mr/Mrs" NAME="Personal Title" />
                    </ROW>
                </RS_DATA>
             </LIST>
          </APPLET>
       </VIEW>
    </SCREEN>
   </APPLICATION>
```

Sample XML Alert Message

This following sample contains a fragment of an XML document output from the Alert Business Service:

```
<?xml version="1.0" encoding="UTF-8" ?>
  <?Siebel-Property-Set EscapeNames="false"?>
- <SiebelMessage MessageId="" IntObjectFormat="Siebel Hierarchical" MessageType="Integration Object"
xmlns="http://www.siebel.com/xml" IntObjectName="Front Office Workflow">
- <ListoffrontOfficeWorkflow>
- <WorkflowProcessDefinition>
  <ActivationDate />
```

```
  <CacheLocalCode>N</CacheLocalCode>
  <Comments />
  <Description />
  <ExpirationDate />
  <Group />
  <LayoutInfo />
  <ProcessBusinessObject>Account</ProcessBusinessObject>
  <ProcessName>SMC Alert - New Main Phone Number Value of Account</ProcessName>
  <ProcessStatus>ACTIVE</ProcessStatus>
  <ProcessType>PROCESS</ProcessType>
  <Runnable>Y</Runnable>
  <TimeLimit />
  <Version>0</Version>
  <ErrorProcessName />
  <PersistenceFrequencyCode />
  <PersistenceLevelCode />
  <ReplicationLevel>None</ReplicationLevel>
- <ListOfWorkflowProcessFlow>
- <WorkflowProcessFlow>
  <ForBranchName>Default</ForBranchName>
  <FromStepName>Get Current Message</FromStepName>
  <FromStepType>TASK</FromStepType>
  <LayoutInfo>@0*0*8*0*0*0*9*LineWidth1*24*FrHS1*24*ToID9*10-5GAS1N2*P06*300
724*NPts1*24*ToHS1*06*FromID9*10-5GAS0H2*P16*384 72</LayoutInfo>
  <ToStepName>Submit Message</ToStepName>
  </WorkflowProcessFlow>
- <WorkflowProcessFlow>
  <ForBranchName>SetFieldValue event</ForBranchName>
  <FromStepName>Start</FromStepName>
  <FromStepType>START</FromStepType>
  <LayoutInfo>@0*0*8*0*0*0*9*LineWidth1*24*FrHS1*24*ToID9*10-5G70RQ2*P06*120
724*NPts1*24*ToHS1*06*FromID9*10-5G70RN2*P16*204 72</LayoutInfo>
  <ToStepName>Get Current Message</ToStepName>
  </WorkflowProcessFlow>
- <WorkflowProcessFlow>
  <ForBranchName>Default</ForBranchName>
  <FromStepName>Submit Message</FromStepName>
```

```
  <FromStepType>TASK</FromStepType>

  <LayoutInfo>@0*0*8*0*0*0*9*LineWidth1*24*FrHS1*24*ToID9*10-5GAS0F2*P06*480
724*NPts1*24*ToHS1*06*FromID9*10-5GAS1N2*P16*564 72</LayoutInfo>

  <ToStepName>End</ToStepName>

  </WorkflowProcessFlow>

  </ListOfWorkflowProcessFlow>

+ <ListOfWorkflowProcessProperty>

+ <ListOfWorkflowStepDefinition>

</WorkflowProcessDefinition>

  </ListoffrontOfficeWorkflow>

  </SiebelMessage>
```

Sample XSL Style Sheets

With no style sheets applied, SWE will return all the requested data. In order to filter data for specific views, it is necessary to apply style sheets for those views and also apply a style sheet that filters data for all unsubscribed views. The following is an example of a style sheet that filters data for all unsubscribed views (*_UnsubscribedView_UI.xsl).

```
<?xml version="1.0" encoding="UTF-8" ?>

- <xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">

  <xsl:output method="xml" media-type="text/html" encoding="UTF-8" />

- <xsl:template match="/">

  <xsl:call-template name="COPYNODE" />

  </xsl:template>

- <xsl:template name="COPYNODE">

- <xsl:copy>

  <xsl:copy-of select="@*" />

  <xsl:value-of select="text()" />

- <xsl:for-each select="*">

  <xsl:call-template name="CHILDPROCESS" />

  </xsl:for-each>

  </xsl:copy>

  </xsl:template>

- <xsl:template name="CHILDPROCESS">

- <xsl:choose>

  <xsl:when test="name()='VIEW'" />

- <xsl:otherwise>

  <xsl:call-template name="COPYNODE" />
```

```
  </xsl:otherwise>
  </xsl:choose>
  </xsl:template>
  </xsl:stylesheet>
```

The following XSL style sheet limits data returned from the SMC Opportunity View (Vendor_SiebelMobileConnector_SMCOpportunityView_UI.xsl):

```
<?xml version="1.0" encoding="UTF-8" ?>
<?xml version="1.0" encoding="UTF-8" ?>
- <xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
  <xsl:output method="xml" media-type="text/html" encoding="UTF-8" />
- <xsl:template match="/">
  <xsl:call-template name="COPYNODE" />
  </xsl:template>
- <xsl:template name="COPYNODE">
- <xsl:copy>
  <xsl:copy-of select="@*" />
  <xsl:value-of select="text()" />
- <xsl:for-each select="*">
  <xsl:call-template name="CHILDPROCESS" />
  </xsl:for-each>
  </xsl:copy>
  </xsl:template>
- <xsl:template name="CHILDPROCESS">
- <xsl:choose>
- <xsl:when test="name()='COLUMN'">
- <xsl:choose>
- <xsl:when test="ancestor::node()[name()='APPLET' and @NAME='Account Form Applet']">
  <xsl:call-template name="AccountFormAppletCOLUMN" />
  </xsl:when>
  </xsl:choose>
  </xsl:when>
- <xsl:when test="name()='FIELD'">
- <xsl:choose>
- <xsl:when test="ancestor::node()[name()='APPLET' and @NAME='Account Form Applet']">
  <xsl:call-template name="AccountFormAppletFIELD" />
  </xsl:when>
```

```
  </xsl:choose>
  </xsl:when>
- <xsl:when test="name()='CONTROL'">
- <xsl:choose>
- <xsl:when test="ancestor::node()[name()='APPLET' and @NAME='Account Form Applet']">
  <xsl:call-template name="AccountFormAppletCONTROL" />
  </xsl:when>
  </xsl:choose>
  </xsl:when>
- <xsl:otherwise>
  <xsl:call-template name="COPYNODE" />
  </xsl:otherwise>
  </xsl:choose>
  </xsl:template>
- <xsl:template name="AccountFormAppletCONTROL">
- <xsl:choose>
- <xsl:when test="@NAME='Name'">
  <xsl:call-template name="COPYNODE" />
  </xsl:when>
  </xsl:choose>
  </xsl:template>
- <xsl:template name="AccountFormAppletCOLUMN">
- <xsl:choose>
- <xsl:when test="@FIELD='Name'">
  <xsl:call-template name="COPYNODE" />
  </xsl:when>
  </xsl:choose>
  </xsl:template>
- <xsl:template name="AccountFormAppletFIELD">
- <xsl:choose>
- <xsl:when test="@NAME='Name'">
  <xsl:call-template name="COPYNODE" />
  </xsl:when>
  </xsl:choose>
  </xsl:template>
  </xsl:stylesheet>
```

B Siebel Mobile Connector Troubleshooting Guide

This appendix lists potential problems associated with setup and implementation of the Siebel Mobile Connector.

Troubleshooting Key

Table 4 lists potential symptoms along with a cross reference to areas in the appendix you should refer to.

Table 4. Troubleshooting Key

For These Symptoms...	Refer to These Questions...
Login error when trying to log in to the Reference Configuration Sample For information on Reference Configuration Sample, see "Using the Reference Configuration Sample" on page 62.	"Question. Are the parameters in the metadata.cfg file set to the correct values?" on page 148
Script time out error when trying to run the Reference Configuration Sample For information on Reference Configuration Sample, see "Using the Reference Configuration Sample" on page 62.	"Question. Have you changed the ASP Script time-out and session time out values?" on page 149
Permission denied error when trying to run the Reference Configuration Sample For information on Reference Configuration Sample, see "Using the Reference Configuration Sample" on page 62.	"Question. Did you change the values of the Execute Permissions and Application Protection options?" on page 149
An error message is returned in XML output when sending XML commands using a Web browser	"Question. Did you receive an error message in XML output when sending XML commands using a Web browser?" on page 149
Cannot upload a file to the Siebel application	"Question. Are you uploading the files programmatically or in a browser window?" on page 149

Table 4. Troubleshooting Key

For These Symptoms...	Refer to These Questions...
Style sheets are being applied, but more data than expected is returned	"Question. Are you trying to use the GetSMCUpdate method to retrieve an update for a particular field of a record?" on page 150 "Question. Are you using the correct spelling and valid values for all commands, methods and arguments to SWE?" on page 150
Cannot query simultaneously for many-to-many relationships between parent and child applets	"Question. Is it possible to query simultaneously for many-to-many relationships between parent and child applets?" on page 150
Alerts not being sent or received	"Question. Are the alerts being received on the destination server?" on page 150

Troubleshooting Questions

1 **Question.** Are the parameters in the metadata.cfg file set to the correct values?

Requirement. The following parameters in the metadata.cfg file must be set with the correct values:

```
GatewayServer=Gateway Server Name
EnterpriseServer=siebel
Port=Port Number
SiebelServer=Siebel Server Name
Language=ENU
```

How to check. Go to *drive*:*install_dir*\siebsrvr\BIN\ENU

Where:

drive = the drive where Siebel Server is installed.

install_dir = the directory where you installed Siebel Server.

Then open the metadata.cfg file and verify that the parameters are set correctly.

2 **Question.** Are the Siebel Server and the desired object managers up and running?

Requirement. Siebel Server and the Siebel Mobile Connector object manager must be running in order to log in or use the Siebel Mobile Connector. The Reference Configuration Sample will not work without the sstchca.dll file.

How to check. To view the object managers:

a Get the OS pid for a particular Object Manager by going to the Siebel Server log directory and opening the siebel.*machine_name*.log file.

b Once the process id is identified, go to Task Manager and look for the SIEBMTSHW.EXE with that PID.

c Go to your registry editor and search for the key value SiebelDataControl.SiebelDataControl.1.

d Verify that this entry exists as a ProgID value and that the InprocServer32 key value for this entry points to the path where the sstchca.dll file is located.

e Make sure that the drive of the file is physical, not virtual. This DLL file is provided with installation of eAI. If this file is not there, then you must reinstall eAI.

NOTE: sstchca.dll is only one of many DLL files required for the Siebel Mobile Connector.

3 **Question.** Have you changed the ASP Script time-out and session time out values?

Requirement. In Microsoft Internet Information Server running on the machine where the Reference Configuration Sample is installed, the ASP Script time-out value must be set to a big number (for example, 2147483646 seconds) and the session time out value must be set to a big number (for example, to 200 minutes).

How to check. In Microsoft Internet Information Server:

a Right-click on the virtual directory created for the Reference Configuration Sample and select Properties.

b Click the Configuration button and click the App Options tab.

4 **Question.** Did you change the values of the Execute Permissions and Application Protection options?

Requirement. In Microsoft Internet Information Server running on the machine where the Reference Configuration Sample is installed, the Execute Permissions must be set to Scripts Only and Application Protection must be set to Low (IIS Process).

How to check. Make sure Execute Permissions is Scripts Only and Application Protection is Low (IIS Process).

5 **Question.** Did you receive an error message in XML output when sending XML commands using a Web browser?

Requirement. The XML Interface to SWE will return an error tag in XML output.

```
<ERROR> we are unable to process your request.
```

This is most likely because you used the browser BACK or REFRESH button to get to this point.

```
</ERROR> if the application does not have a specified home page.
```

You can ignore the error message or, if you prefer, set a home page for the application.

6 **Question.** Are you uploading the files programmatically or in a browser window?

Requirement. Uploading file attachments cannot be done by sending HTTP requests in a Web browser. Instead, uploading files must be done programmatically, so the application that uploads the files can modify the Content-Type of the HTTP request and send the file in the appropriate format for file uploads.

7 **Question.** Are you trying to use the GetSMCUpdate method to retrieve an update for a particular field of a record?

Requirement. The GetSMCUpdate method notes changes that occur at the record level versus for a specific field. If you request an update for a particular field on a record, you will be notified of a change when any field of that record has changed. If your application requires notification about changes to specific fields, then you can use the Alert Business Service for this purpose.

For more information on using the Alert Business Service, see Chapter 6, "Working with the Alert Business Service."

8 **Question.** Are you using the correct spelling and valid values for all commands, methods and arguments to SWE?

Requirement. SWE expects the correct spelling and valid values for all commands, methods, and arguments. Invalid SWE commands, methods, and arguments are ignored. No error message is returned by SWE for such errors and the calling application may experience unexpected results.

9 **Question.** Is it possible to query simultaneously for many-to-many relationships between parent and child applets?

Requirement.

SWE does not currently support querying simultaneously for:

Parent rows in applets (such as all activities)

Child applets (such as all contacts)

Man-to-many relationships between all parent and all child applets.

In other words, it is not possible to perform a query using the XML Web Interface that will retrieve the links between all contacts and which activities they are related to, and vice versa. (such as all contacts related to each activity and vice versa).

There are at least three possible solutions to this task, although the efficacy of these solutions has not yet been firmly established.

First, you can query the database directly for the Siebel table containing the links between activities and contacts, and then map all tables together on the Siebel Server. The disadvantage of this method is that it is database dependent.

Second, you can create a specialized business component using Oracle's Siebel Tools that could perform this task.

Third, you can get the list of children rows for each parent row as you enumerate through the parent rows.

10 **Question.** Are the alerts being received on the destination server?

Requirement. Alert Business Service alerts must be configured to send alerts to a destination server, where the alerts will be handled by a third-party application.

How to check. You may want to create a program residing on the receiving server that outputs the Alert Business Service alerts to a file so that you can view the output. Such a program could work like the following SampleFileReceiver.asp.

```
'--------------------------------------------------------------------
   'SampleFileReceiver.asp
   'Sample receive page that writes document to a file
''--------------------------------------------------------------------
   Option Explicit
   Response.Buffer = True
   Dim objFS, objStream
   Dim PostedDocument
   Dim ContentType
   Dim CharSet
   Dim EntityBody
   Dim Stream
   Dim StartPos
   Dim EndPos
stop
   ContentType = Request.ServerVariables( "CONTENT_TYPE" )
   '
   ' Determine request entity body character set (default to us-ascii)
   '
   CharSet = "us-ascii"
   StartPos = InStr( 1, ContentType, "CharSet=""", 1)
   If (StartPos > 0 ) then
      StartPos = StartPos + Len("CharSet=""")
      EndPos = InStr( StartPos, ContentType, """",1 )
      CharSet = Mid (ContentType, StartPos, EndPos - StartPos )
   End if


   '
   ' Check for multipart MIME message
   '
   PostedDocument = ""
   if ( ContentType = "" or Request.TotalBytes = 0) then
      '
      ' Content-Type is required as well as an entity body
      '
      Response.Status = "406 Not Acceptable"
```

```
    Response.Write "Content-type or Entity body is missing" & VbCrlf
    Response.Write "Message headers follow below:"  & VbCrlf
    Response.Write Request.ServerVariables("ALL_RAW") & VbCrlf
    Response.End
else
    if ( InStr( 1,ContentType,"multipart/" ) > 0 ) then
       '
       ' MIME multipart message. Build MIME header
       '
       PostedDocument = "MIME-Version: 1.0" & vbCrLf & "Content-Type: " & ContentType & vbCrLf & vbCrLf
       PostedDocument = PostedDocument & "This is a multi-part message in MIME format." & vbCrLf
    End if

    '
    ' Get the post entity body

    EntityBody = Request.BinaryRead (Request.TotalBytes )

    '
    ' Convert to UNICODE
    '
    Set Stream = Server.CreateObject("AdoDB.Stream")
    Stream.Type = 1'adTypeBinary
    stream.Open
    Stream.Write EntityBody
    Stream.Position = 0
    Stream.Type = 2'adTypeText
    Stream.Charset = CharSet
    PostedDocument = PostedDocument & Stream.ReadText
    Stream.Close
    Set Stream = Nothing

    '
    ' Write received document to a local file
    '
     Set objFS = CreateObject("Scripting.FileSystemObject")
```

```
      Set objStream = objFS.OpenTextFile("d:\temp\httpoutputfile.txt", 8, True)

      objStream.WriteLine "----------------- Received at " & Now() & " -----------------"
      objStream.WriteLine PostedDocument
      objStream.WriteLine "----------------- End Received at " & Now() & " -----------------"
      objStream.Close
      Set objStream = Nothing
      Set objFS = Nothing

      '
      ' indicate that the message has been received and processed
      '

      Response.Status = "200 OK"
      Response.Write "Done"
      Response.End
   End If
%>
```

B Document Type Definition

This section lists Document Type Definitions for the outbound XML documents Fused with creating subscriptions for use with the Metadata Business Service.

For more information on the Metadata Business Service, see Chapter 5, “Working with the Metadata Business Service.”

DTD for Application Definition

This is the DTD for XML documents passed through the GetApplicationDef method of the Metadata Business Service.

```
<!ELEMENT  APPLICATION          (SCREEN)* >

<!ATTLIST  APPLICATION

   NAME               CDATA            #REQUIRED

   Icon_Caption       CDATA            #IMPLIED

   Node_Id            CDATA            #REQUIRED

>

<!ELEMENT  SCREEN         (VIEW)* >

<!ATTLIST  SCREEN

   Name                CDATA            #REQUIRED

   Viewbar_Text        CDATA            #IMPLIED

   Default_View        CDATA            #IMPLIED

   Node_Id             CDATA            #REQUIRED

   SubscribeForUI      CDATA            #REQUIRED

   SubscribeForUpdate  CDATA            #REQUIRED

   SubscribeForPush    CDATA            #REQUIRED

   UISelectValue       CDATA            #REQUIRED

   UpdateSelectValue   CDATA            #REQUIRED

   PushSelectValue     CDATA            #REQUIRED

>
```

```
<!ELEMENT  vIEW      (APPLET)* >
<!ATTLIST  VIEW
   Viewbar_Text          CDATA            #IMPLIED
   Name                  CDATA            #REQUIRED
   Title                 CDATA            #IMPLIED
   Business_Object       CDATA            #REQUIRED
   Node_Id               CDATA            #REQUIRED
   Visibility_Type       CDATA            #IMPLIED
   SubscribeForUI        CDATA            #REQUIRED
   SubscribeForUpdate    CDATA            #REQUIRED
   SubscribeForPush      CDATA            #REQUIRED
   UISelectValue         CDATA            #REQUIRED
   UpdateSelectValue     CDATA            #REQUIRED
   PushSelectValue       CDATA            #REQUIRED
>
<!ELEMENT  APPLET     (LIST_COLUMN | CONTROL)* >
<!ATTLIST  APPLET
   Name               CDATA            #REQUIRED
   Title              CDATA            #IMPLIED
   Class              CDATA            #IMPLIED
   No_Insert          CDATA            #IMPLIED
   No_Update          CDATA            #IMPLIED
   No_Delete          CDATA            #IMPLIED
   Business_component CDATA            #REQUIRED
   Base_Table         CDATA            #REQUIRED
   Node_Id            CDATA            #REQUIRED
   SubscribeForUI     CDATA            #REQUIRED
   SubscribeForUpdate    CDATA         #IMPLIED
   SubscribeForPush      CDATA         #REQUIRED
   UISelectValue         CDATA         #REQUIRED
```

```
    UpdateSelectValue     CDATA          #IMPLIED
    PushSelectValue       CDATA          #REQUIRED
>
<!ELEMENT  LIST_COLUMN (FIELD) >
<!ATTLIST  LIST_COLUMN
    Name              CDATA            #REQUIRED
    Type              CDATA            #REQUIRED
    Runtime           CDATA            #IMPLIED
    Read_Only         CDATA            #IMPLIED
    Field             CDATA            #REQUIRED
    Table             CDATA            #REQUIRED
    Pick_Applet       CDATA            #IMPLIED
    Calculated        CDATA            #IMPLIED
    Calculation       CDATA            #IMPLIED
    Multivalued       CDATA            #IMPLIED
    Extension         CDATA            #IMPLIED
    SubscribeForUI    CDATA            #REQUIRED
    SubscribeForUpdate    CDATA        #IMPLIED
    SubscribeForPush      CDATA        #REQUIRED
    UISelectValue         CDATA        #REQUIRED
    UpdateSelectValue     CDATA        #IMPLIED
    PushSelectValue       CDATA        #REQUIRED
>
<!ELEMENT  CONTROL     (FIELD) >
<!ATTLIST  CONTROL
    Name                CDATA            #REQUIRED
    Display_Name        CDATA            #IMPLIED
    Type                CDATA            #REQUIRED
    Runtime             CDATA            #IMPLIED
    Read_Only           CDATA            #IMPLIED
```

```
   Field               CDATA            #REQUIRED
   Table               CDATA            #REQUIRED
   Calculated          CDATA            #IMPLIED
   Calculation         CDATA            #IMPLIED
   Multivalued         CDATA            #IMPLIED
   Extension           CDATA            #IMPLIED
   Node_Id             CDATA            #REQUIRED
   SubscribeForUI      CDATA            #REQUIRED
   SubscribeForUpdate    CDATA          #IMPLIED
   SubscribeForPush      CDATA          #REQUIRED
   UISelectValue         CDATA          #REQUIRED
   UpdateSelectValue     CDATA          #IMPLIED
   PushSelectValue       CDATA          #REQUIRED
>
<!ELEMENT  FIELD       (PICKLIST) >
<!ATTLIST  FIELD
   Name            CDATA             #REQUIRED
   Table              CDATA             #REQUIRED
   Type               CDATA             #REQUIRED
   Required           CDATA             #IMPLIED
   Calculated         CDATA             #IMPLIED
   Calculation        CDATA             #IMPLIED
   Multivalued        CDATA             #IMPLIED
   Extension          CDATA             #IMPLIED
   Node_Id            CDATA             #REQUIRED
   Text_Length        CDATA             #REQUIRED
>
<!ELEMENT PICKLIST      EMPTY >
<!ATTLIST PICKLIST
   Name               CDATA             #REQUIRED
```

```
   Node_Id            CDATA            #REQUIRED
   Value              CDATA            #IMPLIED
>
```

DTD for Subscriptions

This is the DTD used for subscriptions to the Metadata Business Service API.

```
<!ELEMENT  SUBSCRIPTIONS (NODE)* >

<!ELEMENT  NODE  EMPTY >
<!ATTLIST  NODE
   NodeId                CDATA            #REQUIRED
   SubscribeForUI        (TRUE|FALSE)     #REQUIRED
   SubscribeForUpdate    (TRUE|FALSE)     #REQUIRED
   SubscribeForPush      (TRUE|FALSE)     #REQUIRED
>
```

Index

Symbols

*** (asterisk), about fields marked by** 95

A

Account Attachment applet, applet details 88
Account Attachment view, view details 82
Account Entry applet, applet details 89
Account Form applet, applet details 83
Activity Attachment applet, applet details 97
Activity Attachment view, view details 93
Activity Form applet, applet details 94
adding records 41
Alert Business Service
- alert output page, printing to a file 76
- HTTP Transport Method, using to set up alerts 75
- SMC alert welcome screen, using to specify default transport method 66
- smcalert.cfg file, configuring 24
- workflow (diagram) 73
- workflow, deactivating 75
- workflow, importing and activating 74
- workflow, list of steps 74
- XML sample document output 140

anonymous user responsibility type, list of responsibilities 27
API methods, table 53
applets, navigating within 37
application definition, configuring
- about 26
- customizing, about 27
- GetApplicationDef, syntax, returns, and example 56
- user responsibilities types (table) 27

Application Representation Metadata Business Service component, about 52
architecture
- components, list of 14
- Metadata Business Service, components, list of 52
- Siebel database, from a third-party mobile application (diagram) 15
- workflow described 16

asterisk (*), about fields marked by 95
authenticating users
- about and architecture 35
- logging in to SWE, about and example 35
- logging off of SWE, about and example 35
- session management, cookies and authentication mode 36

B

Branch Locator
- *See* SMC Branch Locator screen

C

call center representative, list of responsibilities 27
CanInvokeMethod, syntax, returns, and example 54
COM Data Control
- XML output example 29

Contact Attachment applet, applet details 106
Contact Attachment view, view details 100
Contact Form applet, applet details 102
customer, registered, responsibility type 28

D

data sets, retrieving 39
data transfer mechanisms
- authenticating users 35
- real-time access to data 29
- SWE methods, list of 30
- types of 29
- XML commands and SWE methods, about and list of common commands 30
- XML user data, tags used to request user data 31

data, updating and synchronizing
- downloading files, about and procedure 47
- one-way synchronization, about 43
- records, adding 41
- records, deleting 43
- records, modifying 42
- uploading files 44

DDL files, creation location 22
deleting records 43
Document Type Definition
- Metadata Business Service for

subscriptions 159
XML documents passed through GetApplicationDef method 155
downloading files, about and procedure 47
drilling down
about and procedure 38
specific record, about 38
DTD
See Document Type Definition

E

employees
See SMC Employee screen; SMC Responsibility screen
Enable Component Groups dialog box
Alert Business Service, using to 21
Siebel Sales component group, using to select 20
Workflow Management component group, using to select 21
error messages
Reference Configuration Sample, about action 63
eService request
See SMC eService Request screen
Extractor / Parser Metadata Business Service component, about 52

F

field service representative responsibility type, list of responsibilities 27
files
downloading, about and procedure 47
uploading, about and procedure 44

G

GetSMCUpdate method
examples 49
parameters, table of 48
update, getting (procedure) 49
using, about 48
global language support, about 50

H

HTTP Transport Method, using to set up alerts 75
HyperText Transfer Protocol (HTTP), about using 29

I

installation
application definition, configuring 26
default transport mechanisms, setting 25
interactivity mode, setting 23
query, setting the list rows returned 24
server components, configuring 20
server configuration files and DDL files, creation location 22
Siebel component, optional 19
Siebel components, required 19
Siebel Server settings, configuring 22
smcalert.cfg file, configuring and example 24
user agent, configuring 25
interactivity mode, setting 23

J

Java Data Bean
interface, about using 29
Metadata Business Service, about instantiating 54
XML output example 29

L

language, about global language support 50
load subscription screen, about using 67
Load / Save Selections Metadata Business Service component, about 52
LoadSubscription, syntax, returns, and example 58
logging in, authenticating users, and example 35
logging off, about and example 35

M

Metadata Builder Engine Metadata Business Service component, about 52
Metadata Business Service
about 11
API methods for accessing (table) 53
API methods, called in order (steps) 53
architecture components, list of 52
CanInvokeMethod, syntax, returns, and example 54
CreateStyleSheets, syntax, returns, and example 55
customizing the application definition, about 27
Document Type Definition for subscriptions 159
GetApplicationDef, syntax, returns, and example 56
instantiated, about 54
InvokeMethod, syntax, returns, and example 55

LoadSubscription, syntax, returns, and example 58
MetadataCleanup, syntax, returns, and example 59
process flow, steps 52
Reference Configuration Sample user interface 64
Reference Configuration Sample, using 62
SaveSubscription, syntax, returns, and example 60
workflow (diagram) 51
metadata extractor screen
example (diagram) 67
using, about 67
MetadataCleanup, syntax, returns, and example 59
modifying records 42

N

navigating
to a screen 36
within a screen 37

O

Opportunity Attachment applet, applet details 116
Opportunity Attachment view, view details 112
Opportunity Form applet, applet details 113
orders
See SMC eService Order screen

P

partners, creating authentication mechanism 35
predefined queries (PDQ), about and executing 39
printing alert output page to a file 76
pushed alerts, about 29

Q

querying
number of list rows returned, setting 24
performing 38
predefined queries, executing 39
request types 37

R

real-time access to data
about 29
authenticating users, about and architecture 35
dates on records, about parsing and example 40
downloading files, about and procedure 47
drilling down on items 38
hidden fields, retrieving data from 41
large data sets, retrieving 39
navigating to a screen 36
navigating within a screen 37
one-way synchronization, about 43
predefined queries, executing 39
querying items 37
records, adding 41
records, deleting 43
records, modifying 42
uploading files 44
XML commands and SWE methods, about and list of common commands 30
XML user data, tags used to request user data 31
records
adding 41
date and time, about parsing and example 40
deleting 43
large data set, retrieving all records in 39
modifying 42
one-way synchronization, about 43
Reference Configuration Sample
error message received, about action 63
logging off, recommendation for 69
Metadata Business Service component, about 52
style sheets, applying 64
using, about 62
Reference Configuration Sample user interface
load subscription screen 67
metadata extractor screen 67
Siebel applications screen 65
SMC alert welcome screen 66
style sheets screen 71
subscription configuration screen 69
welcome screen 64
registered customer responsibility type, list of responsibilities 28
retrieving data
dates on records, about parsing and example 40
drilling down, about and procedure 38
hidden fields, retrieving data from 41
large data sets, retrieving 39
navigating to a screen 36
navigating within a screen 37
predefined queries, executing 39

querying items, request types 37
querying, performing 38
screens, list of (table) 36
ROW_COUNTER attribute, about using 40

S

sales representative responsibilities, list of responsibilities 28
SaveSubscription, syntax, returns, and example 60
saving, about using SaveSubscription method 60
scenarios
customer service voice scenario 13
sales voice scenario 12
wireless sales scenario 13
screens
list of (table) 36
navigating to 36
navigating within 37
server components, configuring
Alert Business Service, enabling 21
Siebel Sales component group, selecting 20
Siebel Server settings, configuring 22
Workflow Management component group, about enabling 21
server configuration files, creation location 22
Service Request Activity List applet, applet details 121
Service Request Attachment List applet, applet details 122
Service Request Detail applet, applet details 119
Service Request Detail View with Attachments, view details 118
Service Request Detail view, view details 118
session management, about and 36
Siebel applications screen
example (diagram) 65
using, about 65
Siebel COM Data Control
Metadata Business Service, about instantiating 54
using, about 29
Siebel Java Data Bean interface, about using 29
Siebel Mobile Connector Administrator responsibility, list of responsibilities 28
Siebel Mobile Connector, configuring
interactivity mode, setting 23
query, setting number of list rows returned by 24
server configuration files and DDL files, location where created 22
smcalert.cfg file, configuring and example 24
user agent, configuring 25
Siebel Mobile Connector, workflow described 16
Siebel Sales component group, selecting 20
Siebel Server
Reference Configuration Sample, error message received 63
settings, configuring 22
SMC Account Activity List applet, applet details 86
SMC Account Contact List applet, applet details 84
SMC Account Detail - Activities view, view details 81
SMC Account Detail - Contacts view, view details 81
SMC Account Detail - Opportunities view, view details 81
SMC Account Detail - Service Requests view, view details 81
SMC Account Detail view, view details 80
SMC Account List applet, applet details 82
SMC Account screen
about and screen diagram 79
Account Attachment applet, applet details 88
Account Attachment view, view details 82
Account Entry applet, applet details 89
Account Form applet, applet details 83
SMC Account Activity List applet, applet details 86
SMC Account Contact List applet, applet details 84
SMC Account Detail - Activities view, view details 81
SMC Account Detail - Contact view, view details 81
SMC Account Detail - Opportunities view, view details 81
SMC Account Detail - Service Requests view, view details 81
SMC Account Detail view, view details 80
SMC Account List applet, applet details 82
SMC Account Service List applet, applet details 87
SMC Account view, view details 80
SMC Oppty List applet, applet details 85
views available (table) 80
SMC Account Service List applet, applet

details 87
SMC Account view, view details 80
SMC Activity Contact List applet, applet details 97
SMC Activity Detail - Contact view, view details 92
SMC Activity Detail view, view details 92
SMC Activity List applet, applet details
SMC Activity screen 93
SMC Contact screen 108
SMC Activity screen
about and screen diagram 91
Activity Attachment applet, applet details 97
Activity Attachment view, view details 93
Activity Form applet, applet details 94
SMC Activity Contact List applet, applet details 97
SMC Activity Detail - Contacts view, view details 92
SMC Activity Detail view, view 92
SMC Activity List applet, applet details 93
SMC Activity view, view details 92
views available (table) 91
SMC Activity view, view details 92
SMC alert welcome screen
using, about and example (diagram) 66
SMC Branch Locator applet, applet details 129
SMC Branch Locator screen
about and screen diagram 128
SMC Branch Locator applet, applet details 129
SMC Branch Locator view, view details 129
SMC Branch Locator view, view details 129
SMC Contact Account List applet, applet details 107
SMC Contact Detail - Accounts view, view details 100
SMC Contact Detail - Activities view, view details 101
SMC Contact Detail - Opportunities view, view details 100
SMC Contact Detail - Service Request view, view details 100
SMC Contact Detail view, view details 99
SMC Contact List applet, applet details 101
SMC Contact screen
about and screen diagram 98
Contact Attachment applet, applet details 106
Contact Attachment view, view details 100
Contact Form applet, applet details 102
SMC Activity List applet, applet details 108
SMC Contact - Service Request view, view details 100
SMC Contact Account List applet, applet details 107
SMC Contact Detail - Accounts view, view detail 100
SMC Contact Detail - Accounts view, view details 101
SMC Contact Detail - Opportunities view, view details 100
SMC Contact Detail view, view details 99
SMC Contact List applet, applet details 101
SMC Contact view, view details 99
SMC Oppty List applet, applet details 104
SMC Service Request List applet, applet details 106
views available (table) 99
SMC Contact view, view details 99
SMC Employee ReadOnly List applet, applet details 131
SMC Employee screen
about and screen diagram 130
sample XML code 137
SMC Employee ReadOnly List applet, applet details 131
SMC Employee view, view details 131
SMC Employee view, view details 131
SMC eOrder Line Items List applet, applet details 127
SMC eService Order screen
about and screen diagram 125
SMC eOrder Line Items List applet, applet details 127
SMC eService Order view, view details 126
SMC eService Orders List applet, applet views 126
SMC eService Orders List applet, applet details 126
SMC eService Request List applet, applet details 124
SMC eService Request screen
about and screen diagram 123
SMC eService Request List applet, applet details 124
SMC eService Request view, view details 124
SMC eService Request view, view details 124
SMC Opportunity Activity List applet, applet details 115
SMC Opportunity Contact List applet, applet details 114
SMC Opportunity Detail - Activity view, view details 112
SMC Opportunity Detail - Contact view, view details 111

SMC Opportunity Detail view, view details 111
SMC Opportunity List applet, applet details 112
SMC Opportunity screen
about and screen diagram 110
Opportunity Attachment applet, applet details 116
Opportunity Attachment view, view details 112
Opportunity Form applet, applet details 113
SMC Opportunity Activity List applet, applet details 115
SMC Opportunity Contact List applet, applet details 114
SMC Opportunity Detail - Activity view, view details 112
SMC Opportunity Detail - Contacts view, view details 111
SMC Opportunity Detail view, view details 111
SMC Opportunity List applet, applet details 112
SMC Opportunity view, view details 111
views available (table) 110
SMC Opportunity view, view details 111
SMC Oppty List applet, applet details
SMC Account screen 85
SMC Contact screen 104
SMC Responsibility List Administration applet, applet details 133
SMC Responsibility screen
about and screen diagram 132
SMC Responsibility List Administration applet, applet details 133
SMC Responsibility view, view details 133
SMC User List Administration applet, applet views 134
SMC View List Administration applet, applet details 134
SMC Responsibility view, view details 133
SMC Service Request Detail view, view details 118
SMC Service Request List applet
SMC Contact screen 106
SMC Service Request screen 119
SMC Service Request screen
about and screen diagram 117
Service Request Activity List applet, applet details 121
Service Request Attachment List applet, applet details 122
Service Request Detail applet, applet details 119
Service Request Detail View with Attachments, view details 118
Service Request Detail view, view details 118
SMC Service Request Detail view, view details 118
SMC Service Request List applet, applet details 119
SMC Service Request view, view details 118
views available (table) 117
SMC Service Request view, view details 118
SMC User List Administration applet, applet views 134
SMC View List Administration applet, applet details 134
smc.cfg file, configuring 23
smcalert.cfg file, configuring 24
Style Sheet Builder Metadata Business Service component, about 52
style sheets
CreateStyleSheets, syntax, returns, and example 55
Reference Configuration Sample, using to limit data returned 64
sample 142
saved location and naming convention 67
screen, about using and example 71
tip, using to improve performance 64
unsubscribed view, about creating and naming convention 68
subscription configuration screen
example (diagram) 70
using, about 69
subscriptions
Document Type Definition used for 159
load subscription screen, about working with new or existing subscription 67
LoadSubscription method, about 58
SaveSubscription method, syntax, returns, and example 60
SWESetMarkup parameter, about using 47
SWESetRowCnt parameter, about using to retrieve data sets 39
synchronizing records, about one-way synchronization 43
system resources, using MetadataCleanup to free resources 59

T

Third Party Configuration Sample Metadata Business Service component, about 52
ToggleLayout command, using to retrieve

data from hidden fields 41
transport mechanism, setting the default 25
troubleshooting
ASP Script time-out and session time out values, have they changed? 149
correct spelling, using 150
destination server, about receiving alerts 150
Execute Permissions and Application Protection options, have the values changed? 149
files, are you uploading programmatically or in a browser window? 149
GetSMCUpdate method, using to retrieve an update 150
metadata.cfg files, setting the parameters to the correct values 148
querying many-to-many relationships between parent and child, about 150
Siebel Server and object managers, are they up and running? 148
troubleshooting key (table) 147
XML output, receiving error message in when send XML commands 149

U

UNIX systems, server configuration files and DDL files, creation location 23
updates
about 47
GetSMCUpdate method, about using 48
GetSMCUpdate method, examples 49
GetSMCUpdate method, getting update (procedure) 49
uploading files, about and procedure 44
usage scenarios
See scenarios
user agent, configuring for Siebel Mobile Connector 25
user responsibilities, types (table) 27

V

Voice Application capability, setting to TRUE 25

W

welcome screen
example (diagram) 64
using, about 64

X

XML Builder Metadata Business Service component, about 52
XML code, sample 137
XML documents
Alert Business Service sample output 140
DTD for document passed through the GetApplicationDef method 155
XML user tags
common XML tags, list and attributes 32
retrieving data only, about and example 34
retrieving detailed information about the data 35
tags used to request user data, about 31
XML page content, about and tailoring to the application 31
XSL style sheets, sample 142