Siebel Service Handheld Guide

Version 7.8

May 2005

PRODUCT MODULES AND OPTIONS. This guide contains descriptions of modules that are optional and for which you may not have purchased a license. Siebel's Sample Database also includes data related to these optional modules. As a result, your software implementation may differ from descriptions in this guide. To find out more about the modules your organization has purchased, see your corporate purchasing agent or your Siebel sales representative.

Contents

Chapter 4: Application Administration

Chapter 10: Using Service Handheld

Appendix A: Troubleshooting

Index

1 What's New in This Release

What's New in Siebel Service Handheld

Table 1 lists descriptions of the changes made to this version of the documentation to support Release 7.8 of the software.

Table 1. What's New in Siebel Service Handheld, Version 7.8

Topic	Description
"Configuration Guidelines for Siebel Handheld" on page 20	Revised and expanded the general guidelines, including screen and view design, for configuring Siebel Handheld applications.
"Performing Setup for Initial String Translation" on page 33	Revised setup procedure for allowing translation capability in Siebel Handheld applications.
"Configuring Signature Capture Capabilities in Siebel Handheld" on page 38	Revised and expanded this topic with information on enabling signature capture, signature display, and the Signature Capture applet control user properties.
"Creating Siebel Handheld Views" on page 47	Revised procedures for creating new views in the Siebel Handheld applications and adding those views to a user's responsibilities.
"About Thread Throttling" on page 51	Added new topic describing the method for allowing the maximum number of concurrent synchronizations.
"Configuring Application Settings for Siebel Handheld" on page 52	Added new topic describing parameter settings in the Administration - Mobile screen.
"Administering Barcode Settings for Siebel Handheld" on page 57	Revised and expanded this topic with information on enabling barcode scanning, mapping a button for the Socket In-Hand Scan Card, and creating barcode definitions.
"Acting Upon Synchronization Errors in Siebel Handheld" on page 68	Added descriptions of user options after encountering synchronization errors.
Data Filtering "Constructing Business Component and Business Object Filters" on page 70 "Examples of Default Business Object Declarations" on page 71	Revised and expanded Data Filtering chapter with examples of business component filter syntax, business object filter syntax, and default business object declarations.

Table 1. What's New in Siebel Service Handheld, Version 7.8

Topic	Description
Logging Levels "Configuring Direct Server Sync Logging Levels" on page 86	Added information on Logging Level parameters and how to configure them.
"Providing Synchronization Security with Secure Sockets Layer" on page 88	Added information on securing the synchronization data stream by using Secure Sockets Layer (SSL).
"Editing the Handheld Application Installation Configuration File" on page 90	Revised and expanded this table of setup.ini parameters and their descriptions.
"Changing the SyncURL on Siebel Handheld Devices" on page 95	Updated information on using SyncURL to allow communication between the handheld synchronization client and the handheld synchronization server.
"Preventing Synchronization Transaction Conflicts with Siebel Handheld" on page 99	Revised the information on Extended Pick processing and handling Insert Failure processing.
"Direct Server Sync Checklist" on page 140	Revised troubleshooting information to help users determine why or when a synchronization has failed.
"Log Files" on page 141	Revised this section on logs and output files to include information on logging Siebel Handheld synchronization errors.
"SQLTrace" on page 143	Added information on enabling and using SQL tracing.
"PrinterType" on page 187	Revised the list of supported printers for Siebel Handheld.

2 Overview of Siebel Service Handheld

This chapter includes the following topics:

About Siebel Handheld

Siebel Handheld provides a focused subset of functionality to that of the Siebel Web Client. The Siebel Handheld Client differs from the Siebel Web Client and the Siebel Mobile Web Client in that it uses a selected subset of views, accommodates smaller screen sizes, and has a reduced memory capacity. For a list of functional differences, see Chapter 3, "Application Development."

Some features that are available on the Siebel Mobile Web Client are not available on the Siebel Handheld. Scripting and Siebel engines (for example, Pricing and Reporting) are not supported on the handheld. For a list of features that are not supported, see "Unsupported Features for Siebel Handheld" on page 18.

Some supported features of the Siebel Handheld client are as follows:

- **Application Development with Siebel Tools.** Siebel Handheld applications can be configured in Siebel Tools using the Web Applet Designer. You can take your existing Siebel application and use this as the basis for the Siebel Handheld application, or you can create a new application by using the Siebel Tools application shipped on the Siebel Handheld application DVD. See Chapter 3, "Application Development," for more information.
- **Data filtering.** Filtering allows you to specify a subset of data to be used on the handheld device. Filtering is an important feature of a Siebel Handheld deployment because of the relatively large size of the average enterprise database in comparison to the relatively limited memory capabilities of handheld devices. It is important that you allocate sufficient time in the project plan to create and test the handheld filters. See "Application Development" on page 17 for more information.
- **Installation options.** The Siebel Handheld application can be installed on Windows-powered devices in one of two ways. The first is to create a partnership between a desktop or laptop PC and the handheld device using Microsoft ActiveSync. The second is to install the Siebel Handheld application from external media, for example, a Cabinet (CAB) file. Handheld patches can be remotely deployed using PatchAgent. For more information, see Chapter 8, "Installing the Handheld Application."

- **Patch Updates.** Siebel Handheld automatically applies the latest software patches to the handheld device when needed.

  PatchAgent automates the deployment of patches to handheld devices as part of the synchronization process. For more information on PatchAgent, see "Installing the Siebel Handheld Application from External Media" on page 95.

  NOTE: Siebel Service Handheld does not support Companion Sync.

- **Backing Up and Restoring Data.** The Siebel Handheld application provides database backup functionality. A database restore utility is also included with the application. See "Enabling Handheld User Database Backup" on page 93 for details.

Overview of Siebel Handheld Synchronization

The Siebel Handheld Synchronization (Sync) client synchronizes data between the Siebel Handheld database and the Mobile Web Client database or server database. The synchronization process does the following:

- Updates the Siebel server database with changes made to the Siebel Handheld database
- Updates the database on the handheld device with changes made to the Siebel server database
- Downloads metadata changes, such as object definitions for new or modified screens or views
- Allows users to select filters to limit the amount of data that is downloaded
- Automatically applies the latest software patches to the handheld device when needed

Siebel Handheld Synchronization Methods and Architecture

The following methods can be used to synchronize data on the handheld device with data in the Siebel server database: Direct Server Sync (DSS) and Direct Server Sync via Proxy (DSSvP). The architecture of these methods is discussed in the following sections.

Handheld Synchronization

A Handheld synchronization server component run on the Siebel application server can accept concurrent synchronization requests from multiple synchronization clients. When connecting directly to a server, the Siebel Handheld Sync client synchronizes its data directly to a server database.

In Direct Server Sync mode, the Siebel Handheld Synchronization client uses the HTTP(S) protocol over a network connection to communicate with the handheld synchronization server. Users should configure their handheld devices for communication over the desired network.

Direct Server Sync

Direct Server Sync allows users to synchronize their Siebel Handheld application and database directly with the Siebel server. The user connects the handheld device to the Siebel application server through a network connection that uses the HTTP protocol. The connection runs through a IIS Web server to the application server, where the Sync server components are installed and running.

Direct Server Sync via Proxy

Direct Server Sync via Proxy (DSSvP) allows users to synchronize their Siebel Handheld application and database directly with the Siebel server. The user connects the handheld device to the Siebel application server through ActiveSync and the network connection established by the companion PC. The connection runs through a IIS Web server to the application server, where the Sync server components are installed.

Overview of Siebel Handheld Client Application Configuration

The following provides an overview of the configuration process of the Handheld client. Complete the following steps to configure your handheld client application.

To configure the Siebel Handheld client

1. Define the user functionality.

   Determine which user interface elements (such as screens, views, toolbars, applets, find, and help) will be necessary for your Siebel Handheld Client end users. For more information, see "About Defining User Functionality in Siebel Handheld" on page 20.

2. Configure the Siebel Handheld Client user interface.

   The process for configuring user interface elements in the Siebel Handheld Client is the same as that for other Siebel clients. For more information, see "Configuring User Interface Elements in Siebel Handheld" on page 25 and *Configuring Siebel Business Applications*.

3. Identify all eScript or VB script.

   Siebel Handheld applications do not currently support scripting, including eScript or VB script. Scripting used in your Siebel application may conflict with your Siebel Handheld application configuration.

4. Compile the new configuration.

   For more information, see "Compiling the Siebel Handheld Application" on page 32.

5. Test the new configuration.

   Test the configuration, repeating the previous steps, to address any configuration errors.

After completing the Siebel Handheld Client configuration, you are ready to install it on the handheld devices. To install the Siebel Handheld configuration on a handheld device, see the instructions in "Installing the Siebel Handheld Application" on page 105.

3 Application Development

This chapter includes the following topics:

Overview of Siebel Handheld Architecture

The process for configuring a Siebel Handheld Client has a few differences, described in this chapter, from the process for configuring other types of Siebel clients.

NOTE: Readers should familiarize themselves with the Siebel client configuration process described in *Configuring Siebel Business Applications* before attempting to configure the Siebel Handheld Client.

The five layers in the Siebel Handheld architecture are comparable to those used on the Siebel Web Client:

Application layer. The application layer starts and closes the application.

User Interface layer. The user interface layer renders the display and interacts with actions of the user.

Object Manager layer. The object manager layer provides a consistent object behavior and interaction between all business objects within the application.

Data Manager layer. The data manager layer maintains an object-oriented abstraction of the native data stored in the data repositories for the benefit of the object manager.

Database layer. The database layer includes the data that Siebel Handheld users access.

Your Administrator configures an application on the Siebel client using Siebel Tools and compiles the Siebel repository (.srf) file. During the synchronization process, information specific to the Siebel Handheld Client is extracted from the Siebel repository file. This information is used to create a repository metalanguage (.rml) file, which is downloaded to the handheld device.

The repository metalanguage file contains the layout information for the handheld device but not the database layer. The database layer is generated in the transport file called dbfile.txt, which is downloaded and imported to the handheld database (siebelDB.sdf).

For more information on Siebel Tools, see *Using Siebel Tools*.

Unsupported Features for Siebel Handheld

The differences between configuring user interface elements for the Siebel Web Client and the Siebel Handheld Client are shown in Table 2.

Table 2. Configuring Siebel Client User Interface Elements

Client	Siebel Tools	SWT Template Files	Specialized Algorithms
Siebel Web Client	Yes	Yes	No
Siebel Handheld Client	Yes	No	Yes

The following are not supported in Siebel Handheld applications:

- The following applet types are not supported:
  - Chart
  - Explorer
- Group boxes are not supported. See "Group Boxes" on page 26 for more information.
- Alarm Manager is not supported. Therefore, if an alarm check box is added to an applet and the check box is selected, no alarm sounds.
- The base time zone is determined by the system settings on the server and cannot be changed on the handheld device.
- Alphabet tabs are not supported.
- Scripting is not supported. However, any scripts on the business components will be executed during synchronization.
- Siebel Workflow is not supported.

See the following topics which outline, more specifically, other unsupported features.

Multi-Value Groups (MVG)

The Multi-Value Group applet is not supported on the Siebel Handheld Client; however, the Multi-Value Group relationship is supported. The Multi-Value Group (MVG) control button does not appear on fields in Siebel Handheld Client applications. Instead, MVGs can be implemented as parent-child views. As a result, MVG fields are not editable within the parent record. In the Siebel Web Client, addresses of accounts are added, deleted, or edited through the Business Address MVG applet. However, in the Siebel Sales Handheld Client, these functions must be done through a Business Address child applet with an Account applet as its parent.

On the handheld device, all MVG fields are displayed as read-only, with the primary record visible. The first record will be displayed if the primary record is not specified. If you have an application or business requirement that requires the user to view or update all the records in the MVG, you must configure a separate view for that MVG. For example, the Address field in the Accounts or My Accounts view is an MVG. In the Siebel Handheld Client that field is read-only and displays the primary record. In order for the handheld user to see and update the addresses, the new Address view is configured in Siebel Tools. This new Address view is visible in the Show drop-down list.

Functions

Siebel Handheld applications support most of the calculated fields and operations used in the Siebel Web applications with a few exceptions. See Table 3 for a list of the unsupported functions.

Table 3. Unsupported Functions

Function Name
BCHasRows
DockingNodeId
DockingNodeName
DockingNodeRoutId
EAILookupExternal
EAILookupSiebel
EXISTS
GetHQInstanceId
GetHQInstanceName
GetNumBCRows
GetProfileAttr
GetXAval
IsDocked
LanguageName
LocaleName

Table 3. Unsupported Functions

Function Name
LookupMessage
LookupTranslation
NOT
RepositoryId
RepositoryName
RowIdToRowNum
ToolsLanguage

The NOT key word is not supported on Booleans. Use the != operator instead. For example, "Active" != "Y".

The RowIdToRowNum function is supported differently in the handheld applications than in the Web applications. In Web applications, RowIdToRowNum converts a row Id to a numeric value. In the handheld applications, the row ID itself is returned.

About Defining User Functionality in Siebel Handheld

The first step in configuring a Siebel Handheld Client is to determine which user interface elements are necessary. Siebel Handheld Client users have different requirements from Siebel Web Client users. Keep in mind the differences in display size, memory capacity, and input methods between handheld devices and larger computers, such as desktop computers.

The goal is to create applications that are easy for your end users to use and which take advantage of the strengths of the handheld device platform. Because of processing speed, memory limitations, and form factor differences, you should only include those user interface elements that are necessary for users to complete their job responsibilities.

Configuration Guidelines for Siebel Handheld

There are a few general guidelines you should bear in mind when configuring for the Siebel Handheld application. Use these guidelines when you design any new objects for the Siebel Handheld Client. This approach facilitates a logical separation of the Siebel Handheld Client user interface elements from the Siebel Web Client user interface elements.

- Identify User Activities
  - To conserve memory (and thereby improve performance) and ease navigation, identify business processes that are required by end users and develop applications that support these processes.

  - If more than one type of user needs a Siebel Handheld Client application, it is preferable to divide the application into multiple responsibilities rather than give all possible users access to all available screens and views. Responsibilities are fully configurable by the application developer.
  - Do not include additional functionality that is not required to reside on the device.
- Limit the number of screens and views
  - Keep the number of views in your application to 30 or fewer. Determine critical business processes that the handheld will support and only pick the views that are necessary and are compliant with your business requirements.
  - Limit the number of views within a screen to 12 or fewer to keep the View drop-down list concise.
  - Limit the number of screens to 6 or fewer to keep the Screen drop-down list concise.
  - Keep screen names to about 15 characters or fewer, so that they fit in the Screen drop-down list. The number of characters is a general guide because characters vary in width.
  - Keep view names to about 30 characters or fewer so that they fit in the View drop-down list. The number of characters is a general guide because characters vary in width.
- Limit the number of applets for each view

  Design each view so that it has one or, at most, two applets. Limiting the number of applets enhances performance, and enriches the user experience.

  NOTE: Some views contain three applets. Use the toggle button to navigate between the two child applets.
- Limit the number of columns and fields for each applet to minimize scrolling

  Consider the following when designing your applets:
  - Design applications that contain as few screen and view hierarchy levels as possible. In a Web-based application, you may have views with many applets, and the user toggles between the applets. However, for handheld applications, create more views with fewer applets to allow users to quickly find information with a minimal amount of toggling.
  - Include the most important fields and columns that a user needs to see at a glance before scrolling. For example, if you have a form applet that is a parent, include only needed fields to avoid scrolling.
  - If you are using a list applet, determine how wide it needs to be.
  - Design each applet so that it only contains columns and fields that are required for end-user tasks.
  - Consider moving required and editable fields into visible area of the applet where users can enter data and would not have to scroll. Because there is less screen space on a handheld device than on a laptop or desktop, the most important data should be immediately visible.
  - The Date Column can be formatted to show only the date instead of Date and Time where Time is not relevant. This saves the user from having to scroll.
  - To minimize horizontal scrolling, limit the number of columns displayed in a list applet to no more than ten.

- There is no limit on the number of fields in form applets of single-applet views. However, to minimize scrolling, keep the number of fields to 20 or fewer.
- Use a form applet for the parent for each parent-child view.
- In a multi-applet view, limit the number of fields in form applets to five or fewer. Add additional fields only if the field width is short. For example a check box field.
- Reduce the number of fields if the fields are multi-line, for example, a Comments box that contains three lines of text.
- Do not include read-only check boxes in form applets. It is very difficult for users to discern that the check box is not editable.
- Do not add pick applets or set the Runtime property to true for Read-Only fields that have a picklist value associated to them. This will improve overall extraction time, synchronization time and dbfile.txt file size since the process will not be forced to extract all the picklist values for these fields.
- Keep query names to about 15 characters or fewer, so that they fit in the Queries drop-down list. For example, North American Organization is too long for a query name, so you should change it to a shorter name, such as N. American Org. The number of characters is a general guide because characters vary in width (for example, W is wider than i).
- For each screen, create a single-applet More Info view. Create this applet as a form applet. For example, for the Contacts screen, create a More Info view that has the Contacts form applet only.
- For each screen, create a view called My * or All * as a single list applet view. For example, for the Activities screen, create a My Contacts view that has the Contacts list applet only.

- Design Data Filters
  - Limit the size of the dbfile.txt file to less than three megabytes (MB). The RDBMS on the handheld device is approximately three times the size of dbfile.txt. If the data files are so large that they cannot be imported into the database with the available memory, users cannot successfully synchronize their data. See Chapter 5, "Data Filtering" for information on how to design data filters to limit the amount of data that is synchronized to the handheld.
  - When working with assessment creation functionality, you need to name your templates in such a way that they will filter only the needed templates that will be downloaded to the device. Without properly setting up filters, all templates will be downloaded to the device.

About Creating Siebel Handheld Projects

The Siebel Handheld projects appear in the list of projects in Siebel Tools; when Project is selected in the Object Explorer, the list of projects is displayed in the Object List Editor.

Use a naming convention that allows you to easily identify applications, screens, views, and applets belonging to a particular Siebel Handheld application. This allows you to locate all the object definitions in a Siebel Handheld application by querying on the name in the Object List Editor. For example, all of the Siebel Handheld Client screens, views, applets, toolbars, and menus use either an SHCE prefix (for example, SHCE Sales Account List view) or a CE suffix. Use the Object List Editor to query for views that contain the prefix SHCE or the suffix CE; doing so displays all of the Siebel Handheld Client views.

If you create a new Siebel Handheld application, create a new suffix or prefix to identify the name of the application and each screen, view, and applet name. For example, you could use PPC_SHCE for an application.

Designing Screens and Views for Siebel Handheld

Each view in a Siebel Handheld application can display a maximum of two applets at a time, regardless of the number of applets in the view. If there are more than two applets in a view, the user can navigate to additional applets by toggling.

Screen Allocation

The amount of screen space available for applets is determined by the type of applets in the view. Two applets are stacked above one another. A parent form applet in a two-applet view displays multiple fields on a screen. Users use the scroll bar to navigate. The form applet dynamically resizes if there are fewer than five fields and, therefore, does not waste screen display space with empty lines. The maximum number of fields that are displayed at one time is configurable by setting the Max Parent Applet Size preference in the User Preferences dialog. See "Setting User Preferences in the Siebel Handheld Application" on page 118 for more information on this user preference.

Toggling Between Multiple-Applet Views

The Siebel Handheld Client application displays up to two applets at one time. If the view has only one applet, the applet takes up the entire display area. For views with two or more applets, the first two applets in the view are displayed, and the user toggles to see the other applets. The first applet is always displayed and the second applet changes as you toggle.

By default, the user toggles by choosing the View > Toggle menu item. In addition, it is recommended that you provide a toggle button as a visual cue to the end user that there are additional applets. Add a toggle button to each applet except the first applet. Identify the button with a Toggle caption and set methodInvoked to ToggleApplet.

Drill-Down Only Views

Some views may be drill-down only views—that is, they can only be accessed by navigating from another view. These views should not be accessible through the Show drop-down list. However, when the end user navigates to the view, the view title appears in the Show drop-down list. To configure a drill-down only view, set the view title as you would for any other view and set the ScreenMenu property to FALSE. See *Configuring Siebel Business Applications* for more information.

Applet Focus Behavior

When navigating between views, there may be instances when you need to override the default behavior for giving focus to an applet

The GotoView method is used to navigate from one view to another. A button control is added to the applet in the view you want to navigate from, the button's methodInvoked property is set to GotoView and the button's View property is set to the view that is to be created. When the GotoView method is executed, it builds the specified view. If the view includes a parent and child applet, GotoView gives focus to the child applet. This is the desired behavior in most instances when you are simply navigating from one view to another.

If the desired behavior is to navigate to a new view and, additionally, to create a new record, then you must use the GotoViewNewRecord method. This method calls two methods; first it calls GotoView, and then it calls NewRecord, which creates a new record in the new view.

In some instances the default behavior of these methods does not produce the intended behavior. For example, in the Orders view of the Service Requests screen, when you tap the Create New Order button, it takes you to the Order Details view, which contains a parent form applet (Service Request Order) and a child list applet (Service Order Entry Line Item). In this instance, when the Create New Order button's GotoViewNewRecord method is executed, you want a new order to be created. However, the default behavior is to give focus to the child applet rather than creating a new record in the parent applet. Therefore, you need to override the default behavior of GotoView with the control user property OverrideDefaultApplet, and set the value to the applet that you want to give focus. The valid values are 0 (parent) and 1 (child). For example, if you create a button labeled "Create Service Request Order," you would define user properties shown in Table 4:

Table 4. User Properties

Name	Value
View	SHCE Service Order Line Items View
OverrideDefaultApplet	0

Views with Associated Print Templates or Reports

Print templates are text files that you design and create for printing. After creating a print template, you associate it with an applet. When an end user chooses to print from a view, the print template that is associated with the applet is automatically selected and used for printing. Follow these guidelines when creating print templates:

- Only one print template can be associated with an applet.
- The print template name for a particular display applet is registered in the Mail Template property in the Applet object associated with the applet.
- The template must be placed in the Templates directory.
- The template name that goes into the Mail Template property should not include a path name or extension. For example, if the full path of the template is \Program Files\Siebel Handheld\Templates\InvoiceTemplate.txt, you would enter only the root name InvoiceTemplate in the property field.

- The template file itself must have a .txt extension on the device.

For more information, see Appendix F, "Print Configuration Settings."

Configuring User Interface Elements in Siebel Handheld

When configuring user interface elements on the Siebel Handheld Client, also consider the processing speed and memory constraints of handheld devices. The constraint on processing speed affects the performance of a handheld device. The performance of a handheld device is also related to the number of screens and views downloaded to the handheld device. Because handheld devices are not meant to provide the same functionality as larger devices, such as laptops, the number of screens and views must be kept to a minimum on the handheld device.

Due to size constraints on handheld devices, the Siebel Handheld Client displays user interface elements differently from the Siebel Web Client. Consider these differences when configuring the Siebel Handheld Client application. For example, Siebel Handheld Client does not support an alphabetical index.

Configuring List Applets for the Handheld Client

You configure list applets and list columns for Siebel Handheld applications the same way you configure them for other Siebel applications.

Recommended Strategy for Configuring List Applets

The recommended strategy when configuring the Siebel Handheld Client is to use the list applets for record navigation and to rely on form applets to provide the record details.

The following are additional guidelines to follow when designing list applets:

- Configure specific list applets for your Siebel Handheld application, rather than reusing Siebel Web Client list applets.
- Remove all but the essential list columns. Because there is limited screen area to display list columns, omit all unnecessary columns to minimize horizontal scrolling.

  You may either remove the columns from the application or set the Show in List property on the column to FALSE. These two methods are different in the following ways. If the columns are removed from the application, these data won't be downloaded to the handheld device, which keeps the handheld data and a minimum and, in turn, optimizes synchronization performance. If the Show in List property on the column is set to FALSE, the data is still downloaded to the device but is not shown by default. This second approach, though without the synchronization optimization benefit, does allow convenience for the user who wants to customize visible columns. You should take these concerns into account when setting up users.
- Reorder the remaining list columns so that the most frequently used columns are furthest to the left.

- Reduce the default width of the list columns so that more columns can be viewed at one time.
- Use hyperlinks to ease navigation by enabling users to drill into a form applet.

These changes can minimize the amount of horizontal scrolling and column reordering that users must do. Limiting the number of list columns to those that are essential minimizes the amount of data downloaded to the device, resulting in faster synchronization times and more economical use of device memory.

NOTE: All required fields on form and list applets appear with an asterisk (*). If an administrator specifies that a field is required, the field appears with an asterisk in the user interface.

Use Siebel Tools to modify, or rearrange List Columns. For more information on the List Column options, see *Configuring Siebel Business Applications*.

Configuring Form Applets for the Handheld Client

The handheld uses a specialized algorithm to format and display form applets. Due to the display limitations on the device, only one pair of label and field controls are displayed on each row of the form, regardless of the layout shown within Siebel Tools.

Layout Sequence

When configuring for the handheld device, the Applet Web Template Items in the Applet Web Template determine the controls that appear on a form applet. The layout of form applets on the handheld device is determined by the HTML Sequence field set on control objects.

The HTML Sequence field determines the screen layout of a handheld device. Controls are ordered from top-to-bottom based on their HTML Sequence property values. Label and control pairs that do not fit on the same line as other label and control pairs wrap to the next line.

Labels

Keep field labels in applications short (approximately 12 characters or fewer, depending on the character width). Use abbreviations where possible. Labels that are too long are truncated.

Group Boxes

Group boxes are not supported and are not displayed in form applets, even if they are added in the Web Applet Designer. Therefore, reword the field labels to include group box information if necessary. For example, you may have two group boxes on the form, one labeled Ship To and another labeled Bill To. Each group box includes a field named Address. Because group boxes are not supported, the Ship To and Bill To labels are lost, and two fields with the identical Address label remain. Therefore, you must rename the Address labels “Ship To Addr” and “Bill To Addr” or some other label that distinguishes them.

Using Auto Pop-Up Lists in the Handheld Client

Sometimes when a user encounters a new view, he or she must perform an action, such as filling in a required field by selecting a value from a pop-up list. Rather than requiring the end user to tap the control to open the pop-up list, you can configure the view so that the pop-up list automatically opens when the end user gets to that view. To configure a pop-up list to open automatically, add the user property on the applet with Name set to AutoPopupField. Depending on the type of applet, this property is set differently:

- For list applets, set the AutoPopupField property to the Business Component field name.
- For form applets, set the AutoPopupField property to the name of the control.

About Home Page Applets in Siebel Handheld

To configure a Home Page applet, create an applet that has controls of HTML Type Button only. Use the following guidelines in creating your Home Page applets:

- Set the MethodInvoked property of these controls to GotoView. Then add a user property for the control, and set the Name parameter to “View” and set the Value parameter to the view name.
- Set the Caption field.
- Set the HTML Sequence field according to the order in which the buttons are laid out on the device. The buttons appear on the Home Page applet of the handheld device, laid out in rows of three buttons.

About Buttons in Handheld

Set the HTML Type field on buttons to Button, MiniButton, PushButton, or MiniButtonEditNew. All of these HTML Types map to the same button control on the handheld device.

Buttons with text labels on them are sized to the minimum width required to fit the text on the buttons.

NOTE: You cannot use scripting to augment the button functionality in handheld applications.

You may add buttons as long as the method invoked is supported by the class or superclass of the frame or business component. You may also remove buttons from an applet. Be careful when removing buttons because you may alter the behavior of your application in unintended ways. Table 5 describes the behavior of buttons on the different screens. Use this table to guide your decisions about removing buttons from your handheld applications.

For more information of button functionality, see Appendix D, "User Properties and Methods."

Table 5. Button Functionality

Screen	View	Button	Button Functionality
Activities	Activities	Add New Activity	Adds a new activity record.
Activities	Activities	Accept	Accepts an assigned activity. Sets the Status field of the activity to Acknowledged.
Activities	Activities	Decline	Declines an assigned activity. Sets the Status field of the activity to Declined.
Activities	Activities	En Route	Sets the Status field of an activity to In Progress.
Activities	Activities	Arrive	Sets the Start field of an activity to the current time.
Activities	Activities	Suspend	Sets the Status field of an activity to On Hold.
Activities	Activities	Resume	Sets the Status field of an activity to In Progress.
Activities	Activities	Finish	Sets the Status field of an activity to Done, sets the End field to the current time, and sets the % Complete field to 100.
Activities	Recommended Parts & Tools	Check Trunk	Checks each record in the view and verifies the quantity (Qty field) against the inventory. It updates the Available Quantity (Avail) field with the number of items in the end user's trunk.
Activities	Recommended Parts & Tools	Order Part	For any record in the view, if there are insufficient parts that is, Available Quantity is less than Recommended Quantity, this button orders additional parts. The operation generates an Auto Order for the selected products with the Recommended Quantity equal to the quantity in the order line item.
Activities	Time	Start	Creates a new record and sets the start time to the current time.
Activities	Time	End	Sets the end time for the highlighted record.
Activities	Move Parts	Move	Moves a part from a site to your trunk, or trunk to a site.
Activities	Move Parts	Commit	Commits a part movement to the database. Committing a part movement updates the inventories and creates inventory transactions on the server

Table 5. Button Functionality

Screen	View	Button	Button Functionality
Activities	Move Parts	RMA	Generates an order for a part movement and sets the order type to RMA.
Activities	Move Parts	RMA Repair	Generates an order for a part movement and sets the order type to RMA Repair.
Activities	Expenses	New Expense	Adds a new expense record to the Expenses view.
Activities	Invoices	Auto Invoice	Generates an invoice for an activity and creates line items for parts movement, time, expenses, entitlements, discounts, and totals.
Activities	Invoices	Print Invoice	Prints an invoice record.
Activities	Invoices	Sign	Produces a pop-up screen allowing a signature to be recorded.
Service Reqs	Service Requests	Log New SR	Adds a new service request record.
Service Reqs	Orders	Create New Order	Generates a new order record.
Service Reqs	Views	Add Order Line	Adds a new line item to the order.
Service Reqs	Views	Print Order	Prints the selected order.
Service Reqs	Activities	Add New Activity	Adds new activity record.
Service Reqs	Invoices	Auto Invoice	The button executes for each activity in the service request, compiles one total for each line item, and generates one invoice for the entire service request.
Service Reqs	Invoices	Print Invoice	Prints the currently selected invoice onto a handheld printer.
Service Reqs	Invoices	Sign	This allows a person to use the signature capture functionality to sign the generated invoices.
Accounts	Views	Measurem ents	Goes to the Asset Measurements view.
Accounts	Views	Warranties	Goes to the Asset Warranties view.
Accounts	Views	Componen ts	Goes to the Asset Sub-Components view.

Table 5. Button Functionality

Screen	View	Button	Button Functionality
Accounts	Service Requests	Log New SR	Enters new SR record.
Browser	Views	Check Substitute	Checks for substitute parts for the current part. Navigates to the Part Browser Substitutes view.

About the Menu Bar in Handheld

The menu object definition in the Siebel repository that is implemented in handheld applications is named SHCE Generic. The menu object definition in the Siebel repository that is implemented in the Service handheld application is named ServiceCE Generic. The default menu bar configuration for the Siebel Handheld Client includes the following menus: File, Edit, View, and Help. You may remove or rename menus on the menu bar, but you cannot add new menus. You may reorder the menu items by changing the Position property for the menu items. Generally, use the default menu bar configuration for all of your Siebel Handheld Client applications.

About Toolbars in Handheld

This section describes how to configure the toolbar for your application.

Table 6 shows the supported toolbar functions.

Table 6. Supported Toolbar Functions

Toolbar Functions
Back, Forward
New Record
New Query, Execute Query
Next Record, Previous Record, First Record, Last Record
Minimize/Maximize
Delete Record

The default toolbar can be configured in Siebel Tools by configuring the SHCE Main toolbar object. SHCE Main is the main toolbar that is downloaded to the handheld device. If a toolbar named SHCE Main does not exist in your repository, a default toolbar, *Main*, is downloaded instead.

The SHCE Main toolbar contains the default buttons for the device: Back, Forward, New Record, New Query, and Execute Query. Change the order of the buttons on the display by modifying the Position property of the toolbar items. Remove a toolbar button by deleting it or making it inactive.

You can add additional default buttons to the toolbar by creating toolbar items whose Command property matches the name of an active bitmap in the SHCE Command Icons bitmap category. Do not add bitmaps to the SHCE Command Icons bitmap category because only those commands, which are already provided, are supported on the device. No additional commands are supported. You can remove a bitmap or make a bitmap inactive if you do not want it to appear in the Customize Toolbar dialog box.

You can change the bitmap for a toolbar by reimporting the bitmap from the SHCE Command Icons bitmap category.

An end user can personalize the toolbar by selecting View > Toolbar.

For more information on toolbars, refer to *Configuring Siebel Business Applications*.

About the Status Bar in Handheld

The status bar is located above the toolbar. The status bar displays the status of an applet with focus, including the applet title and record item count (for example, Contacts: 1 of 13).

About Hyperlinks in Handheld

Configuring a drilldown, or clicking a hyperlink or dynamic hyperlink in the Siebel Handheld Client, is performed the same way as with other Siebel clients. For more information on configuring drilldowns, see *Configuring Siebel Business Applications*.

Scripting in the Siebel Handheld Client

The Siebel Handheld Client does not execute Siebel Visual Basic (VB) or Siebel eScript scripting.

The Siebel Handheld Client application runs Siebel VB or Siebel eScript scripts and specialized business component logic at synchronization time, rather than in real time on the handheld device. This has significant consequences in applets and controls, because scripts attached to applets and controls are ignored.

CAUTION: Be aware of the following additional issues with respect to scripts and validation:

- **Deactivating scripts.** *Do not* deactivate existing scripts in Siebel Tools to make the handheld device or the synchronization process accept data it is excluding. Errors may occur within the Siebel Web Client or elsewhere where these scripts are run.
- **Calls to user interface methods.** When a business component script makes a call to a user interface method (such as MsgBox), the script cannot be executed, and the record update, deletion, or addition is rejected during synchronization. Strictly speaking, a script attached to a business component should not interact with the user interface.

Any Siebel Visual Basic code should be restricted to the business component level. Applet-level Siebel Visual Basic will not execute and, therefore, should not be written for the Siebel Handheld Client application. As a substitute to using Siebel Visual Basic on the Siebel Handheld Client application, for real-time execution, use the following business component user properties:

- BC/Field Read Only Field...for making fields or business components read-only dynamically.
- Pickmap for fields on the picklist.

Do not use message boxes or prompts in the Siebel Visual Basic code used in conjunction with the Siebel Handheld Client application business components. If the business component is shared between the Siebel Handheld Client application and a laptop, use the following construct to determine if the code is being executed in real time on the laptop or during synchronization:

```
'ActiveViewName is only available when the script is called real time on the laptop

If (TheApplication.ActiveViewName<>"") then

MsgBox"....."

End If
```

- Business component level scripting will be executed during synchronization. Error handling may need to be enhanced if the business component is shared between the handheld application and the laptop.

Compiling the Siebel Handheld Application

Compiling the repository for the Siebel Handheld Client is identical to compiling on the Siebel Web Client. For more information, see *Using Siebel Tools*. For deployment to production, always perform a full compilation. This assumes that you have organized all of your handheld object definitions in one project or a limited set of projects.

When you compile your application, a Siebel repository file (.srf) is created and put in the destination directory you specified during the compile procedure. Make this .srf file the source of repository information for the Siebel Handheld test client machine, defining all Siebel applications for that client, when you move it to your local \Siebel\objects directory.

It is strongly recommended that you make a backup of the existing .srf file in your local \Siebel\objects directory before overwriting it with the new one. That way, if you make an error or you want to revert to the original application, you have a backup that you can easily restore. Versioning for the repository is tracked, and if the .cfg, schema, or repository files have changed, the sync process will automatically rebuild the .rml and schema. For more information, see *Using Siebel Tools*.

Installing and Setting Up End User Translation Functionality

End User Translation functionality provides you with the ability to translate the text strings within the Siebel Handheld application. The translation process should be undertaken during the development cycle and once completed your development team can then package the translated strings for distribution.

The translation process requires the generation of a file of strings from the handheld binaries which are then translated and distributed to the user base. This file, userstr.txt, containing the translated strings is then used to override the strings within the existing handheld binaries. Therefore, you will be required to build a copy of userstr.txt into the installation media for your handheld application.

The translation string file is version specific so you must repeat this process every time you upgrade your Siebel Handheld application.

Performing Setup for Initial String Translation

In order to perform the initial string translation you must perform the following steps prior to installation:

To setup Siebel Handheld application for initial string translation

1. In the Handheld installation directory, add the following two lines to the setup.ini file

   [EndUserLocalization]

   UserDefinedLang=Y

2. Install the Siebel Handheld application.

   During application initialization and the subsequent startup, the file 'userstr.tmp' is generated within the ApplicationDataDirectory (by default this is \Program Files\Siebel Handheld). This file is generated upon startup whenever the UserDefinedFlag is set to Y within the handheld registry and an existing userstr.txt cannot be found.

3. Copy the generated file over to the computer where you will be performing the string translation.

4 Translate the strings as required and remove any untranslated strings from the file.

For example, given the following section of exported strings:

```
33872|HTML\
33873|Tab delimited text file\
33874|Export\
33875|Format:\
33876|Only Active applet\
33877|All applets in view\
33878|Output file name\
33879|Browse…\
33880|And paste output to clipboard\
33881|And open file\
```

Translate the required strings and remove any untranslated strings as follows:

```
33873|Archivo delimitado por tabulaciones\
33874|Exportar\
33875|Formato:\
33876|Sólo el applet activo\
33877|Todos los applets de la vista\
33878|Nombre del &archivo de salida:\
33880|Y pegar salida en el &portapapeles\
33881|Y &abrir el archive
```

NOTE: Strings 33872 and 33879 were removed from the userstr.txt as they were not required to be translated.

5 Save the file as userstr.txt in a UNICODE file format and then copy back into the ApplicationDataDirectory on the handheld device.

6 After restarting the Siebel Handheld application, the translated strings are read from userstr.txt in place of the strings included in the handheld binaries.

Distributing Translated Strings

Once you have generated and tested the userstr.txt file of translated strings, you should include this file in the installation media for the Siebel Handheld application.

To distribute translated strings

Create a new language sub-directory named UserDefinedLanguage within the Siebel Handheld installation directory from which end user devices are installed and which contains the setup.exe.

This directory with the name UserDefinedLanguage will exist alongside the language sub-directories shipped with the application. The new userstr.txt file should be placed in this subdirectory.

Upon installation of the Siebel Handheld application, the custom file will be placed in the ApplicationDataDirectory of the destination device along with the usual handheld files.

Configuring Printing from the Siebel Handheld Application

Users may print from any view in the application. However, they are not necessarily printing what they see on their screen. They may print a portion of the data they view or data that is not viewable at the time.

A print template defines the document that is to be printed. After the template has been defined, it must be associated with an applet in a view.

This section provides information on the following topics:

- Defining documents
- Designing print applets
- Configuring applets for printing
- Creating print buttons
- Creating print templates

Defining Documents for Handheld

Defining documents for printing includes several steps:

- Determine the documents your users need to print.
- Determine the views from which they are most likely to print the documents.
- Lay out the document for printing.

Determining which documents users will need and the views from which they are most likely to print requires that you have a good understanding of the users' day-to-day work.

Whether the existing applets in a view include most of the required data or only a small portion is a secondary consideration. It is not likely that a printout of the existing applets, which are formatted for an electronic PDA interface, will provide an acceptable or usable printed document. In most instances, you need to create additional applets that are specifically used for printing data.

The printed document will be composed of several applets which are added to the view. The applets pull the data from the underlying business components. These applets, in turn, direct the data to the print templates discussed in Appendix E, "Print Tagging Language."

Designing Print Applets for Handheld

The print template references applets associated with the view and is how you specify what data appears in the printed document. For information on print templates, see "Creating Print Templates for the Handheld Application" on page 38. As you design your applets, you should be aware of the print specifications of the printers used in the field. Portable printers typically print on two-inch or four-inch wide paper. You need to design your applets so that the data fits within the limits of the paper width. Keep your documents simple so that they work with different sizes of paper. The following are some guidelines to use when designing your applets to accommodate the smaller paper sizes.

Form Applets

Data from only one business component can be displayed on each line in your document. When you design your form applets, keep in mind that you can display a maximum of two columns of data in form applets. You can specify that each column of data is preceded by an optional caption column. This means you can have up to four columns displayed in your document—two data columns and two caption columns. Alternatively, you may have no captions, and simply display one or two columns of data.

The ratio between caption and data columns is, respectively, 35 percent and 65 percent. This is a fixed relationship that cannot be customized in the template.

Limitations on multiple columns in printed output are largely hardware dependent. In printing 7 fields on a 40 column printer, physical limitations allow only 4-5 characters per field (40 - (7-1))/7 = 4.85. To print more than 4 columns, an 80 column printer is recommended. Generally, the capabilities of impact printers are lower than those of Thermal printers.

List Applets

When specifying the width of a column, you can either specify it with a unit of measurement or as a percentage of the total page width. For example, you can specify that the first column is 20 mm, and the second column is 40 mm. If the width of the paper you print to is narrower than 60 mm, the column widths automatically adjust to fit the printable width as a proportion of the specified width. Therefore, the first column will always be 33 percent, and the second column will be 67 percent of the paper width. The minimum width for a column is 8 mm, which includes a 3 mm gutter margin between columns.

You can specify any number of columns in your list applet. However, if the width of the column is less than 8 mm, the application ignores the column and does not print it.

On narrow paper widths, data from a field may not all fit on one line. Data that does not fit can be specified, in the Applet tag, to wrap to the next line and keep wrapping until the data in that field is completely displayed. The printable width of the page is equal to the paper width minus the left and right margins (PaperWidth – [LeftMargin + RightMargin]). These parameters are set in the setup.ini file. For more information on setting these parameters, see Appendix F, "Print Configuration Settings."

Configuring Applets for Printing for the Handheld Application

Configure print applets as you would any other applet using Siebel Tools. In addition, set the following properties for each applet in the view.

- Set the HTML Popup Dimension value to 0x0.

  The 0x0 setting hides the applet from the end user who should not see the applet that is used to generate the document. Exposing these applets would complicate a product that is tightly configured for an effective and efficient workflow on a small mobile device.

  NOTE: For compatibility with previous versions, Siebel Systems continues to support the name, Popup Dimension, for this property. However, it is recommended that you use HTML Popup Dimension.

- Set the name of the Mail Template to be the name of the print template file. For more information on the print template file, see "Creating Print Templates for the Handheld Application" on page 38. For more information on the Mail Template property, see "Views with Associated Print Templates or Reports" on page 24.

Configuring Print Buttons for the Handheld Application

While users can use the File > Print menu option to print a document, it is helpful to include a Print button on the applet to give users a visual reminder that it is possible to print from a particular view.

When you design your applets, keep the following in mind when deciding where to place the print button:

- Add the print button to the applet that is visible to the user. Do not add the print button to the print applet, which the user never sees.
- Consider what the desired behavior is when determining the placement of a print button.

  For example, in a parent-child view, you may want to add the print button to the parent applet to ensure that the print button is always enabled. This is not necessarily true if the print button is placed on the child list applet. If there are no items in the list applet, the print button is disabled, and the end user cannot print from that view. If you only want the end user to print when there are list items, then placing the button on the child list applet is appropriate.

To configure a print button

1. From Siebel Tools, add a control to the print applet and specify HTML Type = Button.
2. Set Method Invoked = Print.
3. Set the Display Name property of the button. Generally, this is set to Print.

Creating Print Templates for the Handheld Application

Once you have identified and created the necessary views, you are ready to create print templates. A print template is a text file that includes instructions for printing a document. It specifies the data to be printed, document layout options, and text formatting. These instructions are specified using a print tagging language that is described in Appendix E, "Print Tagging Language."

Configuring Signature Capture Capabilities in Siebel Handheld

You can configure the Siebel Handheld Application to capture, view, and print signatures. The following sections provide details on configuring signature capture:

Enabling Signature Capture

Enabling signature capture requires implementing the required supporting field and column definitions, adding a signature control to the applet and, if required, configuring an applet button through which the signature capture dialogue box can be invoked. A field can be configured as a signature control in any list or form applet. Multiple signature controls can be added to a single applet. Once captured, signatures are converted to an ASCII encoding data and stored with the Siebel database as ASCII text.

When enabling Signature Capture, the following basic configuration is required:

- The Business Component field mapped to the signature control should be of Type DTYPE_NOTE
- The underlying Table Column on which the Business Component field is based on Physical Type LONG
- The Applet Control should have a Control User Property called "Signature" with a value "Y". The Control's HTML Type property should be set to "Field"

NOTE: The Applet Control by default does not redisplay the signature data if there is data in the underlying business component field. In this case, you will see the field as read only and will not be able to see the signature that was captured. To redisplay signatures, add another Control User Property called "Show Data" with a value "Y". With this additional configuration, the control is still read-only, but you will be able to see the signature that was captured. For a full list of User Properties available for configuring signature capture, see "Signature Capture Applet Control User Properties" on page 42.

Invoking Signature Capture Dialogue Box and Display

There are two ways to invoke the Signature Capture dialogue box or Signature Display. By default, the basic configuration specified in the Enabling Signature Capture section allows you to invoke the signature capture dialogue box or display a signature using the (...) ellipses button.

You may also use the following basic configuration to invoke the Signature Capture dialogue box or display with the Applet Signature button.

- Create a new Control with properties HTML Type = MiniButton, Method Invoked = StartSignature
- Add a Control User Property to the MiniButton control to indicate the signature field in the business component (for example, the field that is being used to store signature in the business component on which the applet is based).

For examples, see "Configuring Signature Capture in Activities More Info View in Siebel Handheld" on page 40 and "Adding a Signature Capture Button in Siebel Tools" on page 41.

Signature Capture Dialogue Box and Signature Display

The Signature Capture dialogue or display box consists of three controls in the client area: header, signature, and footer. A toolbar displays various buttons depending on the existence of signature data.

Initially, when signature data does not exist or has not yet been captured, the user sees a single line of text within the field control that reads Not Signed. Once the signature capture dialogue box or signature display is invoked, this message appears at the bottom of the screen. The Clear and Cancel buttons are also enabled, while the Accept button is disabled. Once the Signature area within the dialogue box is populated, the Accept button is automatically enabled. Table 7 describes the functionality around these three buttons.

Table 7. Signature Capture Dialogue Box Buttons

Button	Description
Accept	Tap this button to commit the signature to the database.
Clear	Tap this button to reset and clear the signature field.
Cancel	Tap this button to discard the signature and close the display.

Once the signature is committed (for example, signature captured and accepted in the database), the user sees a single line of text "x Signed x" within the applet field control.

The Header and Footer is a line or unlimited number of lines of text in one or more paragraphs. The specific Header or Footer values that appear as well as how they appear in the dialogue box is controlled by a set of user properties. See "Signature Capture Applet Control User Properties" on page 42.

Configuring Signature Capture in Activities More Info View in Siebel Handheld

The following example provides detailed steps to configure signature capture in Activities More Info View in SIA Sales Handheld. The Activities More Info View has one applet 'SIS HH Activity Entry Applet - CE', which is based on the Action Business Component, which in turn is based on S_EVT_ACT Table. The following example shows how to configure a field within the Action Business Component called Signature, which is based on a new column created from S_EVT_ACT_X Table. To enable basic signature capture (for example, no header or footer), follow these steps:

1. In Siebel Tools, add an extension column to the table S_EVT_ACT_X.

For this Field	Add This Value
Name	SIGNATURE
User Name	Signature
Type	Extension
Nullable	TRUE
Physical Type	Long
Length	0
Cascade Clear	Ignore
Transaction Log Code	TRUE
Status	Active
Sequence	999
Comments	Extension Column for Activity Signature Capture

2. Add a field to Action business component with the following values:

For this Field	Add This Value
Name	Signature
Join	X_EVT_ACT_X
Column	SIGNATURE
Text Length	16,383
Type	DTYPE_NOTE
Use Default Sensitivity	TRUE
Comments	Custom Field added for Signature Capture

3 Add control to applet. For this example, you will add the control to the 'SIS HH Activity Entry Applet - CE'.

For this Field	Add This Value
Name	Signature
Caption	Signature
Caption String Reference	SBL_SIGNATURE-1005113048-35Y
Field	Signature
HTML Display Mode	EncodeData
HTML Row Sensitivity	TRUE
HTML Type	Field
Runtime	TRUE
Text Alignment	Left
Visible	TRUE
Comments	Custom control added to display and invoke Signature Capture.

4 Add two Control User Properties to the new 'Signature' control with the following values:

User Property Name	Value
Signature	Y
Show Data	Y

5 Add the new control to the Edit Web Template for 'SIS HH Activity Entry Applet - CE'.

You can now invoke the signature capture dialogue box by tapping on the ellipsis (...) on the signature field on the applet.

Adding a Signature Capture Button in Siebel Tools

Optionally, you can also configure the signature capture dialogue box by placing a button on the applet. The following configuration steps provide details on doing this.

To create a new minibutton control

1 Create a new MiniButton control against the 'SIS HH Activity Entry Applet - CE'.

For this Field	Add This Value
Name	SignatureCapture
Caption	Sign
Caption String Reference	SBL_SIGN-1005113047-35W
HTML Type	MiniButton
MethodInvoked	StartSignature
Comments	Custom MiniButton to Invoke Signature Capture.

2 Add a Control User Property to the MiniButton control to indicate the Signature field in the business component:

- Name = SignatureField
- Value = Signature

Signature Capture Applet Control User Properties

The user properties, shown in Table 8, are optional and provide you with the ability to modify the style and layout of the Signature Capture applet.

Table 8. Style and Layout Modification Properties

User Property	Description	Sample Value
Signature	If Y the field becomes a signature field	Y
ShowData	If Y, the signature data can be redisplayed. Setting this User Property to 'N' disables the ellipses in the signature applet control, when there is signature data already existing	Y or N
EncryptKey	Encrypt key string. If this is not specified, an internal mechanism is used to generate the encryption key.	1 and 99999999
WritingGuide	Defines appearance of writing guide in signature control.	Y or N
BorderStyle	Defined the border style of the signature control.	Double
InkColor	Defines the ink color. Uses a Hex format (OxBBGGRR) of blue, green, red.	OxFF0000 (Blue)
InkWidth	Defines ink width	2 (Pen width)

The user properties shown in Table 9 are added to support header and footer display and the save function.

Table 9. Header and Footer Display Properties

User Property	Description	Sample Value
SignatureDataField	This property is used to specify a destination field for the captured signature in scenarios where it differs from the current UI field. When specified, it specifies the name of the business component field that the signature data is fetched from and saved to.	SigDataField
SignatureHeaderField	Specifies the name of the BC field from which the header value is fetched.	SigHeader
SignatureFooterField	Specifies the name of the BC field from which the footer value is fetched.	SigFooter
SignatureHeaderSaveField	Specifies the name of the BC field to which the header value is saved.	SigHeaderSave
SignatureFooterSaveField	Specifies the name of the BC field to which the footer value is saved.	SigFooterSave
MaximumHeaderLines	Specifies the maximum height of the header display control in number of text lines. If the header text exceeds this number, the user needs to scroll down the control to view the rest of the header text. (Default = 4).	4
MaximumFooterLines	Specifies the maximum height of the footer display control in number of text lines. If the header text exceeds this number, the user needs to scroll down the control to view the rest of the header text. (Default = 4).	4
SignatureHeight	Specifies the height of the signature capture control by pixels (default = 80, max = 200).	Integer between 1 and 200.
SignatureDate	If set to Y, the current date is appended below the last line of the footer. The date string is saved to footer save field.	Y
SignatureTime	If set to Y, the current time is appended below the last line of the footer to the right of date if date is specified. The time string is saved to footer save field.	Y
ConfirmAccept	If set to Y, a confirmation message is displayed after accept button is pressed. Otherwise, no confirmation message is displayed.	Y

NOTE: If the content of the Header or Footer exceeds the MaximumHeaderLines or MaximumFooterLines, respectively, then the content can be scrolled vertically using the pen. No scrollbar will be generated.

Testing the Siebel Handheld Application

Before deploying your application to your end users, be sure to thoroughly test the application by synchronizing to a handheld device. The handheld configuration is downloaded to the device where you can test the functionality and verify that you have the desired behavior. Another equally important aspect of handheld testing is to check to see how much data is downloaded to the device. If too many records are downloaded, this increases the length of time it takes to synchronize the handheld device and slows down the application performance. Check the server log file to see how many records are downloaded for each business component. You must turn on logging to capture this information in the log file.

For more information on enabling logging for Direct Server Sync, see "Configuring Direct Server Sync Logging Levels" on page 86.

4 Application Administration

This chapter includes the following topics:

Overview of Siebel Handheld Administration Screens

This chapter is designed to show you how to administer Siebel Handheld. Key tasks are outlined that you can perform using the Administration - Mobile screen. Your organization may follow a different task sequence according to its business requirements.

Following are the views used for administering Siebel Handheld applications. You can access the Administration - Mobile screen through either the Siebel Web Client or Siebel Mobile Web Client.

- **Session Administration:** This view contains a detailed information about user synchronization sessions. Use this view to monitor and investigate handheld user synchronization activities.
- **Server Component Administration:** This view allows you to manage server-level settings for synchronization components. These settings are used by the synchronization engine to extract application definition and user data. These settings will override the application-level settings.

- **Application Administration:** This view lists all the handheld applications available within your enterprise. From this view you can create new application settings and administer settings that have already been created. The contents of the following views are used by the synchronization engine to extract application definition and user data.
  - **Business Component Filters:** Use this view tab to change synchronization filter settings for business components. If you wish to restrict the contents of selected business components to reduce the data set otherwise visible to your users, you can change filter settings in this view.
  - **Business Object Filters:** Use this view tab to define business object level filters, which restrict data that is downloaded to the handheld client upon synchronization.
  - **Settings:** Use this view tab to define parameter settings for your application.
- **Conflict Administration:** This view allows you access selected transaction errors.
  - **Audit Trail:** Use this view tab to obtain information about action taken on a particular data conflict. Audit Trail must first be enabled for the conflict data. The business components are Handheld Transaction Error and Handheld Txn Error Message.
  - **Business Components:** Use this view tab to see business component information associated with a particular data conflict.
  - **Error Details:** Use this view tab to get detailed information about specific data conflicts. You are also able to take particular action on any given conflict.
- **Barcode Administration:** This view allows you to create new barcode definitions for your handheld applications.
- **User Administration:** This view allows you to manage user-specific settings, such as business object and business components filters, that filter the data that display to the user. The data sets contained in these user views are used by the synchronization engine to extract application definitions and user data. The Siebel Handheld application is populated by these data when the user performs synchronization. These settings override the server component-level and application-level settings.
  - **User Business Components:** Under this view tab, you can set user-specific sync filter directives for business components. These settings override application-level settings.
  - **User Business Objects:** Under this view tab, you can set user-specific filters for business objects. In this setting, you limit the data set that is downloaded, at the time of sync, to the Handheld client. These settings override application-level settings.
  - **User Settings:** Under this view tab, you can control user-specific settings which override server component-level and application-level settings.
- **Barcode Enabling:** In this view you can enable barcode scanning for chosen views in the Siebel Handheld application.

Setting Up Users in Siebel Handheld

To set up users on a Siebel Handheld application, perform the following tasks:

- "Assigning User Responsibilities to Siebel Handheld Users" on page 47

Assigning User Responsibilities to Siebel Handheld Users

You must create user accounts and assign responsibilities and passwords for each handheld user. The steps are the same as those for creating user accounts, assigning responsibilities and passwords for the Siebel Web Client. See *Applications Administration Guide* for more information on this topic.

Creating Siebel Handheld Views

The handheld synchronization engine extracts views based on configuration (SRF) and the responsibility set up in the Application Administration view.

To configure a new view in your handheld application, follow the list below. Knowledge of Siebel Tools is required to perform these tasks.

1. Create the new view in Siebel Tools. For information on creating views, see *Configuring Siebel Business Applications*.
2. Add that view to the handheld screen. For more information on adding views to screens, see *Configuring Siebel Business Applications*.
3. Add the screen to the handheld application if required. For more information on adding screens to applications, see *Configuring Siebel Business Applications*.
4. Compile the project. For information on compiling projects, see *Using Siebel Tools*.
5. Add the new view to the Siebel Handheld user's responsibility. See "To add a handheld view to a user's responsibility" on page 47.

To add a handheld view to a user's responsibility

1. Launch the Web Client application that uses the same database that the Siebel Handheld application synchronizes with.
2. Choose Navigate > Site Map > Administration-Application > Views.
3. Add the new views.
4. In the Responsibilities view, associate each view with the appropriate handheld responsibility.
5. Add appropriate users to the responsibility.

   The predefined responsibility for Siebel Service Handheld is Field Technician - Handheld.

   NOTE: For more detailed instructions on adding views and associating views with responsibilities, see *Applications Administration Guide*.

Specifying Views for the Handheld Application

The Siebel Handheld application uses a small subset of views used in the Siebel enterprise application. Because there are memory resource constraints on handheld devices, it is recommended that you compile an SRF to extract only those views that are used on the device.

The combination of the SRF and the user's responsibility determines which views are extracted. The handheld device only extracts views whose parent screen is included in the handheld application and are part of the user's responsibility.

About Business Component Filters

Business component filters are used in some cases to restrict the size of extractions that otherwise might be excessive. They suppress the extraction of business component records. Business component filters should be employed with discretion because excessive filtering can cause server data integrity problems.

Business component filters can be specified at either the Application level or the User level, but not at the Component level. They should not be specified at the user level outside of your test environment, and once tested should be migrated to the Application level.

If you are configuring for handheld synchronization, a suggested practice, is to start with no business component filters at all. Attempt a sync and even if the sync fails due to excessive data extraction volumes, view the extraction results to determine the business components for which the data extraction was excessive. If any business component extraction is found to be excessive, you can reduce the extraction by adjusting the existing search specs specified in Siebel Tools (applets and business components), or Handheld Settings (for example, in DefaultBusObj), and selection filters (business object filters). If nothing else works, a business component filter can be used to force a reduction in the result set for a particular business component.

Setting Business Component Filters for Siebel Handheld Users

The Business Components view in the Application Administration screen contains the sync filter settings for business components. If you wish to add or remove business component filter settings for users, you do so in this view.

To add a business component filter setting

1. From the application-level menu, select Navigate > Site Map > Administration-Mobile > Application Administration.
2. In the Application Administration screen, select the appropriate application, then click the Business Component view tab.

3 Click New, then enter the required information.

Some fields in the Business Component view are described in the following table:

Field	Comments
Owner	Read only. Contains the Siebel Handheld application to which the business component will be associated. For example, Siebel <Application name> for CE.
Business Component	A text box. Enter the business component name.
Name	A text box. Enter the business component filter name. For example, Filter 1.
Query	A text box. Enter the syntax for the query. For example, [Start Date] <=Today() For more information on developing queries, see "Data Filtering."

To remove a business component filter setting

1 From the application-level menu, select Navigate > Site Map > Administration-Mobile > Application Administration.

2 In the Application Administration screen, select the appropriate application, then click the Business Component Filters view tab.

3 Select the filter setting record wish to remove.

4 From the Business Component Filters menu, choose Delete Record.

About Business Object Filters

The Siebel Web client supports named queries on most or all screens. These queries restrict the result set displayed to the user. Business object filters are the Handheld Sync equivalent to these named queries. Just like client based queries, they are applied to a specified business object. They also contain search or sort specifications for business components that are part of that business object.

The Default column is the only difference from the query syntax appearing in a PDQ filter for the Siebel Web client. The Default column is unique to the business object filter mechanism. For Web Client queries, users can set a default query for each view in their Preferences screen. If users do not do this, there is no default query, so all records are returned by default. This is not recommended for handheld deployments as this would result in excessive data extraction volumes. Thus, the Default column allows you to decide what filter should be used by default if a user does not make a selection. There is also a client option to revert to default filters selection. There should be at most a single default filter for each business object name. It is not recommended that you set more than one filter as the default for a particular business object. Note however, that the same default choice will be made by all clients.

Business object filters can be specified at either the Application level or the User level, but not at the Component level. Business object filters specified at the User level will override a filter specified at the application level provided the application-level filter has the identical values for the business object name and filter name. The list of filter choices presented to the user is the merged list of all application filters except those overridden at the user level, plus all user filters.

Setting Business Object Filters for Siebel Handheld Users

The Business Object Filters View in the Application Administration screen contains the sync filter settings for business objects. If you wish to add or remove business object filter settings for users, you do so in this view.

To add a business object filter setting

1. From the application-level menu, select Navigate > Site Map > Administration-Mobile.
2. In the Application Administration screen, select the appropriate application, and then click the Business Object Filters view tab.
3. Click New, and then enter the required information.
4. The fields in the Business Object Filters view are described in the following table:

Field	Comments
Owner	Read only. Contains the Siebel Handheld application to which the business component will be associated. For example, Siebel <Application Name> for CE.
Default	Check to allow you to decide what filter will be used by default if a user does not make a selection.
Business Object	A text box. Enter the business object name. For example, Account.
Name	A text box. Enter the business object filter name. For example, BOFilter 1.
Query	A text box. Enter the syntax for the query. For example, 'Account'.Search = "[Account Status] = LookupValue (""ACCOUNT_STATUS"", ""Active"")" 'Account'.Sort = "Name, Location". For more information on developing queries, see "Data Filtering."

To remove a business object filter setting

1. From the application-level menu, select Navigate > Site Map > Administration-Mobile > Application Administration.

2. In the Application Administration screen, select the appropriate application, and then click the Business Object Filters view tab.
3. Select the filter setting record wish to remove.
4. From the Business Object Filters menu, choose Delete Record.

About Thread Throttling

When a large number of users attempt to synchronize simultaneously, the load on a single server can become excessive. Thread throttling significantly reduces the total number of simultaneously active transaction processor threads and database extraction threads, thereby increasing the overall throughput for synchronization, without reducing the number of concurrent users. Synchronization performance degrades slightly but will successfully complete for all users.

Thread throttling allows the maximum possible number of concurrent users. It throttles in a manner that maximizes application server usage during any mixture of transaction processor and database extraction threads. Thus, if the server supports *x* concurrent transaction processing threads, it will allow this number of transaction processing threads while disallowing all database extraction threads. Similarly, if the server supports *y* concurrent database extraction threads, it will allow this number of database extraction threads while disallowing all transaction processing threads. If fewer than *x* transaction processing threads are currently active, some database extraction threads are allowed to execute concurrently. Similarly, if fewer than *y* database extraction threads are currently active, some transaction processing threads are allowed to execute concurrently.

For information on thread throttling settings, see "Configuring Application Settings for Siebel Handheld" on page 52.

To enable thread throttling, see "To add a new application setting" on page 56.

About Settings in the Handheld Administration Screens

Settings can appear at three levels: Application, Component, or User. Certain settings are only allowed at specific levels.

There are also two types of settings: single-line and multi-line settings. For a single-line setting, only one row may contain a unique setting name at a given level. Thus, a single-line setting at the User level may only be used once at that level. For single-line settings, the Sequence column should always contain a value of zero. For multi-line settings, the sequence column is necessary to allow multiple rows. The sequence column has no effect on actual processing, but is part of the logical key in the database table. When you create a new record in this database table, the record must have a unique logical key. For single-line settings, you may leave the sequence column value alone, which will result in a default value of zero being assigned. For multi-line settings, each line must include a different sequence number to avoid logical key conflicts. If the combination of setting name and sequence number is not unique for a particular parent record (Application, Component, or User), you will receive an error message when stepping off or saving the record. Multi-line entries must contain correct the key values before they can be saved to the database.

The rules for override are that any setting, whether single line or multi-line, specified at the lower levels overrides that same setting name at higher levels. Thus the settings at the application level apply unless specific settings are overridden for a particular component or a particular user. When an override occurs, it occurs for the complete setting name. Thus, if a particular multi-line setting has three rows at the user level and twelve rows at the application level, the resulting multi-line setting value for that setting will contain only the three rows at the user level. There is also an override mechanism to allow a setting to be removed, rather than replaced. To remove a setting from a higher level, specify the setting name to be removed, with an empty setting value.

Configuring Application Settings for Siebel Handheld

The Administration-Mobile screen contains a Settings list that allows you to change default settings. Some of the settings are described in Table 10.

The Handheld setting value field has a limit of 2000 characters. To create a default business object (DefaultBusObj) setting value with more than 2000 characters, use the following general approach:

Break the value into multiple lines of DefaultBusObj settings. These settings should all be entered using the same DefaultBusObj value format:

- BusObj|Visibility|Query|BusComp_1|query specifications_1[|BusComp_n|query specifications_n]

When defining multiple lines of DefaultBusObj settings for a single business object, all view modes and filters should match. If a business component is found in multiple DefaultBusObj settings for a single business object, then the business component's query specifications will be concatenated with a space and appended into the previous business component's query specifications. You should be aware of this in order to decide if both query specifications should be combined with an OR or AND operative.

The DefaultBCuserProps and DefaultFields settings for a single business component can also be broken into multiple lines of settings if their values exceed 2000 characters. These settings should all be entered using the same value format as well:

- BusComp|BCUserProp_1[|BCUserProp_n] or BusComp|Field_1[|Field_n]

Table 10. Application Settings

Setting Name	Definition	Valid Values
AllowRememberPassword	Single-line setting. Specifies whether the Remember Password check box will be displayed on the Handheld Client Synchronization screen.	TRUE or FALSE
ApplicationDirectory	Single-line setting. Specifies the working directory (UNC path) of the Siebel Handheld synchronization component where user data are stored.	\\server name\directory name

Table 10. Application Settings

Setting Name	Definition	Valid Values
DBExtractThreadLoad	Single-line setting. Enables Thread Throttling functionality for database extract threads. It is the load (or weight) that will be assigned to each individual Database Extractor thread. For each running database extracting thread, this load value will be used to determine the total system load, which is then compared against the MaxTotalThreadLoads value. A new thread will only be started if the total system load does not exceed the MaxTotalThreadLoads value. A value of zero disables throttling of Database Extractor threads.	integer >= 0; default is 2
DefaultBCUserProps	Multi-line setting. Reduces the number of User Properties that are extracted to the device metadata, thus decreasing the number of fields for a business component that need to be extracted. These reductions lower the metadata extract size and optimize client performance by removing irrelevant user properties from the handheld configuration. If this directive is not specified for a business component, then all user properties will be extracted.	
DefaultBusObj	Multi-line setting. Used to force the extraction of a business object and its designated business components. This is typically used when a business component is not extracted because it is not specified via the "DefaultView" directive and is required for the application to function correctly. Another use of this directive is to designate business components that function as RBOs (reverse business objects).	Business Object Name|Visibility Level||Business Component Name|QBE query construct For more on QBE (query by example) structure, see "Constructing Business Component and Business Object Filters" on page 70.

Table 10. Application Settings

Setting Name	Definition	Valid Values
DefaultFields	Multi-line setting. Used to force the extraction of fields for a particular business component. This is typically the case where the business component is extracted via the "DefaultView" directive but because the field is not visible on the applet, it is not extracted. Also when a hidden business component is extracted via the "DefaultBusObj" directive, it is necessary to specify all of the fields that need to be extracted for the business component. If no fields are specified via this directive, then the business component will not be extracted for this case.	Business Component Name|Field1|Field2|…
DefaultHandheldInsertMode	Single-line setting. Enables insert failure processing for synchronization. If enabled, synchronization will stop upon encountering a data conflict during an insert operation. End users or administrators must resolve the data conflict before synchronization can resume. This feature is disabled by default.	NoFail or FailOnError
DefaultHandheldSyncPickMode	Single-line setting. Enables Extended Pick processing for synchronization. If enabled, synchronization will stop upon encountering a data conflict during a pick operation. End users or administrators must resolve the data conflict before synchronization can resume. This feature is disabled by default.	NoFail or FailOnError
EnableBatchExtraction	Single-line setting. Enables or disables Batch Sync functionality.	TRUE or FALSE
EnableThrottleDebugMode	Single-line setting. This parameter enables or disables Thread Throttling functionality.	TRUE or FALSE
MaxThrottleWaitTime	Single-line setting. For Thread Throttling functionality. This is the upper limit on the amount of time (in minutes) any individual throttling request will be allowed to wait.	integer >= 0; default is 5

Table 10. Application Settings

Setting Name	Definition	Valid Values
MaxTotalThreadLoads	Single-line setting. For Thread Throttling functionality. It defines the maximum number of total thread loads (determined by DBExtractThreadLoad and TranProcThreadLoad for each type of thread) that will be allowed to run across the scope defined by ThreadThrottleScope. If a throttling request is made that causes this maximum to be exceeded, the requesting thread is delayed. The default value is zero, which completely disables the throttling mechanism.	integer >= 0; default is 0
NoDatabaseBusComp	Multi-line setting. Defines a business component that must be omitted from the device database and excluded from database extraction.	
NoExtractBusComp	Multi-line setting. Defines a business component that must be excluded from database extraction.	
SchemaChangeNotification	Multi-line setting. Defines a Business Service that should receive the Handheld Schema Change Notification.	(Siebel internal)
ThreadThrottleScope	Single-line setting. For Thread Throttling functionality. It defines the scope of the thread throttling mechanism as using one of two different modes: Process Thread (throttling occurs independently within each process), or Application (throttling occurs across all processes running the same handheld sync application on a single server). Only the first character of the ThreadThrottleScope parameter will be used to determine the value. All other characters are ignored. Thus “aardvark” is interpreted as “Application.” Case is also ignored. The default value is “Application.”	Application or Process

Table 10. Application Settings

Setting Name	Definition	Valid Values
TranProcThreadLoad	Single-line setting. For Thread Throttling functionality. It is the load (or weight) that is assigned to each individual Transaction Processor thread. For each running transaction processing thread, this load value is used to determine the total system load, which is then compared against the MaxTotalThreadLoads value. A new thread will only be started if the total system load does not exceed the MaxTotalThreadLoads value. A value of zero disables throttling of transaction processing threads.	integer >= 0; default is 3
TranProcThreshold	Single-line setting. For Thread Throttling functionality. This is the minimum time (in minutes) that the Transaction Processor will continue processing transactions before performing a handshake with the client. Once this value has been exceeded, a handshake occurs instead of starting to process the next transaction. Doing so prevents the client from timing out the session when thread throttling is enabled.	integer >= 0; default is 5

To add a new application setting

1. From the application-level menu, select Navigate > Site Map > Administration-Mobile > Application Administration > Settings.
2. From the Settings list menu, select New Record.
3. Click in the name field and select the desired application setting.
4. Click in the Value field and enter the text of the setting.

Configuring Component Level Settings for Siebel Handheld

You administer server components for your handheld device from the Administration-Mobile > Server Component screen. The component displayed is based on the application selected in the Application Administration view. The Server Component Administration screen is used for overriding application settings at the component level on a specific server. The data entered here is used by the synchronization engine when the user synchronizes with the server.

The Server Component screen contains a list of server components associated with the application that was selected in the Application Administration screen. For each server component, the Settings view details the associated settings.

The Name field value needs to be the exact component name, such as "SalesceObjMgr_enu", of the Handheld Sync server component.

Server component settings are used primarily for functionality required for individual server components. For example, thread throttling is dependent on components running on a particular server and is best set at the server component level. Examples of settings that might be made at the component level are MaxTotalThreadLoads, TranProcThreadLoad, DBExtractThreadLoad, and so on. Generally, settings made at the application level are shared by all server components.

Setting up Inventory Locations in Siebel Service Handheld

Be sure that any organization that your end users belong to is included as an organization of the External Location inventory location. This is required for the Siebel Service Handheld application to work correctly.

Administering Barcode Settings for Siebel Handheld

Siebel Handheld can scan and parse data from HIBC and UCC/EAN barcodes. Siebel Administrators can set up Siebel Handheld to scan and parse any one-dimensional barcode and render the corresponding data in an applet. For example, if a control in a list applet is enabled for barcode scanning and the barcode scanned is valid, the handheld application will parse the barcode, create a new record, and fill in the field with parsed data. In form applets, after the user scans an item, the data representing the item automatically populates the barcode-enabled field. In order to do this, Siebel Administrators must define the barcodes and how data will be parsed from these barcodes in the Barcode Administration view.

Enabling Barcode Scanning in Siebel Handheld

Before using the barcode scanning feature, you must add the following parameter value in setup.ini:

```
EnableBarcode = Y
```

NOTE: In the event that your handheld application is not preconfigured to enable barcode scanning, you must make an Application Event Services entry in Siebel Tools.

To enable barcode scanning

1. Navigate to Application object, and then select Siebel Service for CE.

2 Expand to Application Event Services object, add a new entry with the following:

Name: SHCE Core Barcode Service
Event Name: HHCE_Start
Business Service: SHCE Core Barcode Service

If you are going to scan barcodes, you need to associate the scan trigger with a button on the handheld device. This task may not apply to some rugged devices because those devices may have an embedded scanner and a specific button to scan barcode.

Mapping a Button for Socket In-Hand Scan Card

Siebel Handheld applications support embedded barcode scanners provided in Symbol and Intermec devices. In addition, Siebel Handheld applications support certain Socket scan cards. See *System Requirements and Supported Platforms* on Siebel SupportWeb for a list of supported handheld devices and Socket scan cards. Note that different Socket scan cards will require different scan card driver programs to be installed. Check with the card vendors/websites for required driver program installation.

For Intermec and Symbol devices, the barcode scanning can be triggered by the built-in buttons. If you are using Socket scan cards, the following steps are required to be taken on the Handheld device for mapping a button to the barcode-triggering utility software, SocketTrigger.exe.

To map a button to SocketTrigger.exe (Socket Scan Card only)

1 Navigate to Start > Settings > Personal > Buttons.

2 From the Button list, select a button to trigger In-Hand Scan Card.

3 In the Button Assignment field, select SocketTrigger, and then tap OK.

Creating New Barcode Definitions for the Handheld Application

You can define new one-dimensional barcodes in the Barcode Administration views. To define a new barcode, you need to create a new barcode definition in the Barcodes list, then create corresponding records in the Barcode Item list to define data elements within the barcode.

To create a new barcode definition

1 From the application-level menu, select Navigate > Site Map > Administration-Mobile > Barcode Administration.

2 Click New, then fill in the fields.

The fields in the Barcodes list are described in the following table:

Field	Comments
Name	Required. Any descriptive name for the barcode. Must be unique for all records defined in the Barcode screen.
Sequence	Required. This is the sequence for barcode parsing. The lower of two sequence numbers takes precedence when deciding which barcode in the list to match. For example, if a scanned barcode matches two entries in the barcode list, the one with a lower sequence number will be picked for the match. The lowest sequence number is 1.
Barcode Type	Required. Type of barcode standard. Can either be Location or Code. For a Location type barcode, the barcode data element parsing is done solely by the Sequence field in Barcode Item list. For a Code type barcode, the barcode data is first parsed into multiple sections by the Code field in the Barcode Item list, then is further parsed by the Sequence field for each section. See "Example of Location Type Barcode" on page 61 and "Example of Code Type Barcode" on page 62.
Buscomp	Optional. Used to designate into which business component the target data is placed.
Min Length	Optional. The minimum length for the barcode data in digits. If it is not set, the default minimum length is 1.
Max Length	Optional. The maximum length for the barcode data in digits. If it is not set, the default maximum length is 10000.
Prefix	Optional. Defines the value that the parsing algorithm uses to identify barcode definition to use. The allowed format for Prefix is any ASCII character for exact match, '#' for any single digit number match (that is, 0 - 9), '&' for any alphabet character in lower and upper case, or '*' for a wild card match. For example, you can enter ##9&* as the prefix, which means the first two characters can be any number, the third character has to be '9', the fourth character can be any alphabet character, and the fifth character can be any ASCII character.
Suffix	Optional. Allows you to associate suffixes with individual fields. The allowed format for Suffix is the same as the one specified for Prefix. In addition, the rules for matching Suffix are the same as the ones for Prefix.
Description	Optional. Detailed description for this barcode.

3 In the Barcode Item list, add new records to identify how data will be parsed from barcode.

Depending on the type of barcode standard you just defined (either Location or Code), you then need to create new records in the Barcode Item list to identify how data is parsed.

4 In the Barcode Item list, Click New, then update the required fields.

Some of the fields are described in the following table:

Field	Comments
Name	Required. Name of the barcode item (for example, Product Name corresponds to the Product Name field)(for example, Product Name corresponds to the Product Name field). This name does not determine which field in BusComp the parsed data is written to. That field is determined by the "Data Desc" field.
Code	Applicable for Code barcodes only. Used to break a barcode into multiple barcode sections.
Sequence	Required. The sequence for barcode data elements parsing. For Location barcodes, this corresponds to the sequence used to parse data elements from the barcodes. For Code barcodes, this corresponds to the sequence used to parse data elements within barcode sections.
Min Length	Required. Minimum Length of the character string for this data element. If this is a variable length string, then Minimum Length is not equal to Maximum Length.
Max Length	Required. Maximum Length of the character string for this data element. If this is a variable length string, then Minimum Length is not equal to Maximum Length.
Data Desc	Optional. Used by algorithm to identify what field to parse this data into. For example, Product Name corresponds to the Product Name field.
Data Format	Optional. Defines format for data element (for example, MMDDYY is for date).
Action	Optional. Defines the action for the data element. Both Data Format and Action are written to the field specified in "Data Desc" by the following format: "<parsed data element>,<Data Format>,<Action>". See the example in Table 14.

Example of Location Type Barcode

The two entries for the Barcode list, HIBC Secondary and IBM, are shown in Table 11.

Table 11. Location Type Barcode List Entries

Name	Sequence	Barcode Type	Buscmp	Min Length	Max Length	Prefix	Suffix	Description
HIBC Secondary	1	Location	Medical Product		31	+$$9#### ###3		
IBM	2	Location	Tech Product		30	+$$9#### ###39		

Entries for Barcode - HIBC Secondary in the Barcode Item list are shown in Table 12.

Table 12. Location Type Barcode Item List Entries

Name	Code	Sequence	Min Length	Max Length	Data Desc	Data Format	Action
Prefix		1	4	4	Prefix		
Quantity		2	5	5	Quantity		
Exp date flag		3	1	1	Exp flag		
Date		4	6	6	Expiration Date	YYMMDD	
Lot		5	0	13	Lot #		
Link		6	1	1	Link		
Check		7	1	1	Check		

If the barcode being scanned is +$$90010039509283C001LZ, both barcode entries "HIBS Secondary" and "IBM" are good matches. Since HIBC Secondary has a lower sequence as 1, it is picked as the final match for parsing the barcode. The data is parsed as follows:

{(Prefix, "+$$9") (Quantity, "00100") (Exp flag, "3") (Expiration Date, "950928,YYMMDD") (Lot #, "3C001") (Link, "L") (Check, "Z")}

NOTE: In this example, the use of "Prefix" in the two tables is different. The Prefix column in the Barcode list table is required for matching the record in the Barcode list. The Prefix row in the Barcode Item list table is a user-defined data element to be parsed for a particular barcode.

Example of Code Type Barcode

Entries for Code type barcode in the Barcode list are shown in Table 13.

Table 13. Code Type Barcode List Entries

Name	Sequence	Barcode Type	Buscomp	Min Length	Max Length	Prefix	Suffix	Description
UCC/EAN	1	Code	Medical Product		37	01		

Entries for Code type barcode in the Barcode Item list are shown in Table 14.

Table 14. Code Type Barcode Item List Entries

Name	Code	Sequence	Min Length	Max Length	Data Desc	Data Format	Action
AI 01	01	1	2	2	AI 01		
Packaging Indicator	01	2	1	1	Indicator		
UCC/EAN #	01	3	12	12	UCC/EAN		
Link	01	4	1	1	Link		
Quantity Section	30	1	3	5	Quantity Section	1000	Validate
AI 240	240	1	3	3			
Product #	240	2	1	13	Product #		

The barcode UCC/EAN is parsed to three sections, 01, 30, and 240. If the barcode being scanned is 0**1**10312345678903**30**50**240**ABCDEFG, the data is parsed as follows:

Parsed Data = {(AI 01, "01") (Indicator, "1") (UCC/EAN, "031234567890") (Link, "3") (Quantity section, "3050,1000,Validate") (Product #, "ABCDEFG")}

NOTE: The Code portions in the barcode appear in bold text.

Enabling Applications Views for Barcode Scanning

You can enable any existing or new view for barcode scanning.

To enable an application view for barcode scanning

1. From the application-level menu, select Navigate > Site Map > Administration-Mobile > Barcode Enabling.
2. Click New, then enter the necessary information.

   The fields in the Barcodes list are described in the following table:

Field	Comments
View	The view in the handheld application from which the scan occurs.
Applet	The applet within the view from which the scan occurs.
Buscomp	The business component from which information is drawn.
Buscomp Field	The specific field within the business component that should be updated through barcode scanning.

Example: Process of Creating a New Barcode Definition in the Service Handheld Application

Although this example is for the Siebel Service Handheld device, it pertains to all Siebel Handheld products. You can define new one-dimensional barcodes in the Barcode Administration views. To define a new barcode, you need to create a new barcode definition in the Barcode list, then create corresponding records in the Barcode Item list to define data elements within the barcode.

To create a new barcode definition, complete the following procedures:

- "To create a new barcode definition for Siebel Service Handheld" on page 63
- "To enable the SHCE Service FS Activity Parts Movements views for barcode scanning" on page 64

To create a new barcode definition for Siebel Service Handheld

1. From the application level menu, select Navigate > Site Map > Administration-Mobile > Barcode Administration.
2. Click New, then fill in the necessary fields.

   The following table describes the fields in the Barcodes list.

Field	Sample Data
Name	Hard Disk 100 GB
Sequence	Set to the greatest number in the current list view plus one
Barcode Type	Location
Buscomp	FS Activity Parts Movement

Field	Sample Data
Min Length	12
Max Length	12
Prefix	#
Description	Service

3 In the Barcode Item list, add a new record to identify how data will be parsed from the barcode.

The following table describes the fields in the Barcode Item list.

Name	Sequence	Min Length	Max Length	Data Desc
UPC Code	1	1	1	
MFG Code	2	5	5	
Product Name	3	5	5	Product Name
CC	4	1	1	

NOTE: Reducing the amount of barcode seed data extracted to the Service Handheld application improves client performance and reduces synchronization database extract size.The following steps provide an example of extracting only the barcode standards that are required for the Service Handheld application. This is applicable as long as all the barcode records for the Service Handheld application defined in the Barcode list have the Description attribute set to "Service."

4 From the application level menu, select Navigate > Site Map > Administration-Mobile > Application Administration.

5 In the Application Administration view, select “Siebel Service for CE.”

6 Click the Settings view tab, and make sure the parent record is the application “Siebel Service for CE.”

7 Locate the "DefaultBusObj" directive that has the following value:

- "SHCE Barcode|0||SHCE Barcode||SHCE Barcode Item||SHCE Barcode Enable|".

Replace that value with the following value:

- "SHCE Barcode|0||SHCE Barcode|[Description]="Service"|SHCE Barcode Item||SHCE Barcode Enable|".

To enable the SHCE Service FS Activity Parts Movements views for barcode scanning

1 From the application-level menu, select Navigate > Site Map > Administration - Mobile > Barcode Enabling.

The Barcode Enable list appears.

2 If the following record does not exist, create a new record to enable the Product Name field in the SHCE Service FS Activity Part Movements List Applet for barcode scanning with the following information:

View	Applet	Buscomp	Buscomp Field
SHCE Service FS Activity Part Movements	SHCE Service FS Activity Part Movements List Applet	FS Activity Parts Movement	Product Name

3 If the following record does not exist, create a new record for the Product Name field in the SHCE Service FS Activity Part Movements Form Applet for barcode scanning with the following information:

View	Applet	Buscomp	Buscomp Field
SHCE Service FS Activity Part Movements (More Info)	SHCE Service FS Activity Part Movements Form Applet	FS Activity Parts Movement	Product Name

4 Save the changes. Now the Buscomp Field "Product Name" in both the "SHCE Service FS Activity Part Movements" view and the "SHCE Service FS Activity Part Movements (More Info)" view have been enabled for barcode scanning. The Buscomp Field "Product Name" specified here matches the value in the Data Desc column in the Barcode Item list defined earlier.

NOTE: The example provided above assumes that the product Hard Disk 100GB has been defined and properly set in the appropriate product catalog.

About Monitoring Synchronization Sessions for Siebel Handheld

You monitor Siebel Handheld application synchronization sessions and synchronization conflicts from a number of screens within the Siebel Mobile Web Client, or if you are using DSS or DSSvP you do so by way of the Siebel Web Client. Every time a user synchronizes with the server, a record is made.

The following topics relate to synchronization monitoring.

- "Monitoring Synchronization Sessions for Siebel Handheld" on page 66
- "Obtaining Error Details from Siebel Handheld Synchronization Sessions" on page 66
- "Obtaining Business Component Information from Siebel Handheld Synchronization Conflicts" on page 67

- "Obtaining an Audit Trail for Siebel Handheld Synchronization Sessions" on page 67
- "Acting Upon Synchronization Errors in Siebel Handheld" on page 68

Monitoring Synchronization Sessions for Siebel Handheld

The Session Administration view allows you to gather all data related to each sync session a user performs. The Sync Status list appears above two child lists: Sync Sessions and Extraction Info. These lists allow you to do the following:

- Sync Status: Allows you to monitor each user sync session, and includes such data as: sync status, the date, type of sync and so on.
- Sync Sessions: Allows you to gather further information on the status of each sync session, including conflicts during the sync, the size of the data extracted, and so on.
- Extraction Info: Allows you a detailed look at all data extracted during the sync session, including business component information, the number of records extracted. You use this list to help get answers to questions such as the following:
  - Why did a particular record download to the device?
  - Why is an expected server record not downloaded to the device?
  - Why is the total number of records showing up in some view so big?
  - Why is the total number of records showing up in some view so small?
  - Why does the extraction take so long?

    There is one extraction record for each unique extraction context. The extraction record will list the filters used to control the extraction, the context of the extraction, how many records exist in that context for those filters, and how long it took to extract that set of records. By analyzing these records, and looking at all extraction changes resulting from filter changes, and, or changes to search specs (defined in Siebel Tools), you can answer all of the above questions.

To monitor synchronization sessions

- From the application-level menu, select Navigate > Site Map > Administration-Mobile > Session Administration.

  The Sync Status list appears, with the two child lists, Sync Sessions and Extraction Info, below it.

Obtaining Error Details from Siebel Handheld Synchronization Sessions

The Conflict Administration view allows you to obtain details of sync errors that occur during individual sync sessions.

To obtain synchronization error details

1. From the application-level menu, select Navigate > Site Map > Administration-Mobile > Conflict Administration.
2. In the Transactions list, select the appropriate record, and then click the Error Details view tab.
3. In the Error Details list, select or query for the desired record.

Obtaining Business Component Information from Siebel Handheld Synchronization Conflicts

The Transactions view on the Conflict Administration screen allows you to obtain business component information related to synchronization errors that occur during individual sync sessions.

To obtain business component error details

1. From the application-level menu, select Navigate > Site Map > Administration-Mobile > Conflict Administration.

   The Transactions view appears.
2. In the Transactions view, select the appropriate record, and then click the Business Components view tab.

Obtaining an Audit Trail for Siebel Handheld Synchronization Sessions

The Transactions view on the Conflict Administration screen allows you to obtain audit trail information for an action that was taken against a particular transaction.

NOTE: You must enable Audit Trail separately. For information on enabling Audit Trail, see *Applications Administration Guide*.

To obtain audit trail information for synchronization error details

1. From the application-level menu, select Navigate > Site Map > Administration-Mobile > Conflict Administration.

   The Transactions view appears.
2. In the Transactions view, select the appropriate record, and then click the Audit view tab.

Acting Upon Synchronization Errors in Siebel Handheld

Upon reviewing a synchronization error, you can act upon it by performing one of three different actions.

To act upon handheld sync transaction errors

1. From the application-level menu, select Navigate > Site Map > Administration-Mobile > Conflict Administration.

   The Transactions list appears. The transaction buttons are described in the following table:

Action	Result
Reapply	Reapplies the selected transaction with values listed in the view (or Field Value view). These values override the results provided by the server.
Export	Exports the content of the transaction in the same format used by the sync engine. This output file can be sent to Siebel Technical Support for problem analysis.
Ignore	Accepts the current status and does not act upon the error. The record becomes read-only. There is no impact, except that the error will no longer appear in queries for unresolved conflicts.

2. Click either Reapply, Export, or Ignore depending on your needs.

5 Data Filtering

The following topics are covered in this chapter:

Developing Data Filters for Handheld

The following general steps are recommended for developing Handheld synchronization filters.

- Identify the screens and views of your application. Then identify the business components that are required to support each screen and view, noting the areas where each business component occurs. For example, identify the form, list, or pick applets that are based on each business component.
- Within Siebel Tools, establish the base parent-child relationships and any reverse relationships at the component level.
- Review the business components that occur in picklists and consider creating business component filters for those that will cause large numbers of records to synchronize to the device. These filters are created and maintained in the Mobile Administration screens. For each business component, determine whether it is possible to apply a search specification to all instances of the business component within the Siebel Handheld application. Develop a set of suitable business component filters in the Mobile Administration screens.

  Common examples of business components that are suitable for global business component filters include:

  - Employees (for large organizations)
  - Activities (for example, filter for current Activities, Activities of only certain kinds)
  - Accounts (for example, by geography)
  - Products (for example, active, orderable products)
  - Orders (for example, open, closed in last three months)
- Create appropriate business object synchronization filters for each business object and make sure that each business object includes a default filter. Default filters make sure that users avoid accidental download of an excessive number of records.

Constructing Business Component and Business Object Filters

Business component filters are used to limit the extraction of a business component for every context in which it is extracted.

Example of Business Component Filter Syntax:

Business component filters use query by example (QBE) syntax.

```
[Status] = "Not Started" OR [Status] = "In Progress" OR ([Status] IS NULL) OR [Status] = "On Hold" OR
([Status] = "Done" AND [Done] >= Today() - 7)
```

For more information on QBE syntax, see *Siebel Developer's Reference*.

Business object filters are used to supply filters that a user can select at sync time. These are the filters that are displayed in the SetFilters dialog on the device.

Example of Business Object Filter Syntax:

Business object filters use predefined query (PDQ) syntax.

PDQ syntax rules are a follows:

- The business component name should be a valid business component in the business object on which the PDQ is based
- Search and Sort can be specified for all active buscomps in the business object on which the view is based, appended to each other in any order using space as delimiter.

The normal search spec and sort spec syntax applies except that quotes in the search spec have to be escaped using double quotes since the search spec itself is enclosed in double quotes.

The syntax is as follows. Descriptions of the terms are shown in Table 15.

```
'Account'.Search = "[State] = LookupValue ( ""STATE_ABBREV"", ""CA"" )" 'Account'.Sort = "Name, Location"
```

Table 15. Business Object Filter Syntax

Syntax Term	Description
Account	business component name
Search	key word
"[State] = LookupValue (""STATE_ABBREV"", ""CA"")"	search specs using PDQ syntax
State	field name
LookupValue	function for find an LOV value
STATE_ABBREV	LOV type
CA	LOV value
'Account'.Sort = "Name, Location"	sort spec for sorting the returned list

For more information on PDQ syntax, see *Siebel Developer's Reference*.

Business Object Filters

All Siebel Handheld business object filters are available for selection by a user at the time of sync. Additionally, there is no naming convention for business object filters.

A default filter for a business object is indicated by checking the Default flag in the Administration-Mobile > Application Administration > Business Object Filters view of the Siebel Web client. Be sure to define a default filter for each business object in the Siebel Handheld application. If no filters are applied during the synchronization process, an unacceptably high volume of data will be downloaded to the handheld device.

CAUTION: You should select only one filter as the default for each business object. If two filters are marked as the default for the same business object, the actual default can be determined only by attempting to sync and selecting default filters from the sync client filter selection UI.

Recommendations for Managing Handheld Business Object Filters

The following section provides recommendations for creating and managing filters according to synchronization methods.

Direct Server Sync Deployments

For Direct Server Sync deployments, administrators can create user-specific business object and business component filters in the Administration-Mobile > User Administration view. For more information, see "Setting Business Component Filters for Siebel Handheld Users" on page 48 and "Setting Business Object Filters for Siebel Handheld Users" on page 50.

Both application-level and user-level filters can be used in this deployment. They are managed in the same set of views. Changes to any filters will be effective on your next sync session, without having to restart the Handheld synchronization OM component.

These business object filters are made available to all end users and, therefore, must be defined with the needs of all end users in mind.

Examples of Default Business Object Declarations

You can set default business object declarations in the Administration-Mobile > Application Administration > Settings screen, as described in the following examples.

Example 1

"State Model" default business object setting:

```
State Model - Engine|0||State Model|[BusComp Name]="Expense"|State Model -
Transition||State Model - State||State Model - Position|
```

The first three segments of the declaration are followed by a series of repeating BusCompName Search Spec pairs:

1. Business object name
2. Visibility Type
3. (Siebel internal)
4. Root business component name
5. Search spec QBE for root BusComp records
6. Child business component name
7. Search spec QBE for child BusComp records (this should normally be empty, causing all children to be extracted)
8. Grandchild business component name
9. Search spec QBE for grandchild BusComp records (this should normally be empty, causing all grandchildren to be extracted)

   and so on

The total number of fields must be at least 5 and must be an odd number.

Example 2

Action|0||Action||Action Employee|

This example default business object declaration results in an extraction of the Action and Action Employee business components where they occur with visibility 0 using the business object filter indicated. The business object filters are generally used only for extracting business components that do not correspond to any views or applets. Business object declarations identify specific business components within the business object to be extracted. The business object filters normally give a query based on the primary business component in the business object and force extraction of any business components mentioned in the query when there is no corresponding applet within the application that can cause the desired extraction to occur.

Example 3

In the following example, the declaration is used to extract all the Price List Line Items under the Order Entry (Sales) business object.

```
DefaultBusObjs13=  Order Entry (Sales)|0||Price List Item|
```

Note that this declaration lacks an associated QBE clause. When this is the case, all of the records available to the user for the given visibility are extracted.

Exercise caution when modifying a default business object setting for pick applets, and do so only with the support of a skilled Siebel configurator.

Use default business object declarations to force the extraction of data for a given business component in the context of a given business object and Visibility type. For more information on visibility designations, see "Designating Visibility" on page 73. The default business object extractions are performed in the first phase of data extraction.

You will most often use these declarations to extract additional data that is not extracted as part of the user interface extraction. This may be useful for business components that are referenced programmatically by other business components, but are not exposed in the user interface. The default business object declarations are used to enforce the extraction of a subset of data to the handheld that would not have been otherwise extracted or, most commonly, to filter the data downloaded for dynamic picklists or pick applets.

All pick data is not downloaded to the device due to the memory constraints on the device. Only pick data for fields that are editable is downloaded. Therefore, if you are doing a query on a pick field, you may not be able to display the picklist. When viewing a record in an applet, if a pick field is read-only, its pick button will be disabled. If end users select a read-only field with a pick control, they receive the following error: "No data available in picklist because field is read-only."

Designating Visibility

It is very important that you use the correct visibility designation in the default business object declarations. The visibility entered should be the visibility that governs the data displayed for the business component within the context of the business object specified in the declaration. In the previous Example 1, visibility code 3 is designated. This corresponds to All Visibility, which is the visibility that applies to the Account business component when it is used in a pick applet.

You determine the correct visibility to use by examining View and business component properties within Siebel Tools. When creating a default business object declaration to restrict the volume of data extracted for a pick applet, begin by examining the Popup Visibility Type in Business Component Properties. Cross-reference the visibility type to the list below, and enter the appropriate visibility code in the declaration. If the Popup Visibility Type field on the business component is null, the default value is All. Table 16 summarizes popup visibility types and the corresponding visibility codes.

Table 16. Visibility Codes

Popup Visibility Type	Visibility Codes
Sales Representative	0
Manager	1
Personal	2
All	3
None	4
Organization	5
Contact	6

NOTE: Do not use the Group, Catalog, or Sub Organization Popup Visibility Types. These are not supported in the CE handheld applications.

Overriding the Popup Visibility

If the Popup Visibility Type setting is not sufficiently restrictive (for example, All Visibility), this can lead to a large number of records being extracted when the picklist is processed during synchronization. Making the Popup Visibility Type more restrictive reduces the number of records extracted for picklists. The OverridePopupVisibility user property for an applet overrides the Popup Visibility Type on a business component. Use this property when the desired visibility differs from the business component's Popup Visibility Type, and you do not want to change the Popup Visibility Type. If several picklists use the same business component, you can also use the OverridePopupVisibility user property to vary the visibility of the picklists in the different applets.

There are two considerations you should keep in mind if multiple applications are sharing the same repository. First, changing the Popup Visibility Type has a global effect for all instances where the business component is used. If multiple applications share the same repository and use the same business components, changing the Popup Visibility Type for a common business component could have an undesirable effect for another application. Second, if multiple applications share the same applet, the picklist in each applet shares the same visibility as defined by OverridePopupVisibility property. This is because the OverridePopupVisibility property is set on the containing applet.

In Siebel Tools, navigate to Applet > Applet User Properties, and define the OverridePopupVisibility user property on the applet containing the control that opens the pick applet. See Table 17 for a description of the syntax. Assign a value to this property using the syntax shown, where <buscomp field n> is the underlying business component field of an applet control or column, and <visibility type> is the code corresponding to the desired visibility of the picklist's business component. (See Table 16 for a list of the visibility codes.) Because the name of the OverridePopupVisibility parameter must be unique, append a unique number to the name of the parameter for any other picklists, for example, OverridePopupVisibility1, OverridePopupVisibility2, and so on.

Table 17. OverridePopupVisibility Syntax

Name	Value
OverridePopupVisibility	<buscomp field 1>, <visibility type>
OverridePopupVisibility1	<buscomp field 2>, <visibility type>
OverridePopupVisibility<n>	<buscomp field n + 1>, <visibility type>

NOTE: If OverridePopupVisibility is used, then the picklist and pick applet search specifications are ignored during synchronization when the records for the picklist are extracted.

Siebel Service Handheld Filters

The purpose of the handheld data filters is to identify a select amount of data that is downloaded to the device.

Before you begin designing your own filters or customizing the default business component and business object filters, it is recommended that you read this section to understand the preconfigured filters. They are designed to provide optimal synchronization and application performance.

In addition, see Chapter 10, "Using Service Handheld" to see how the preconfigured application works and how the relationship between the business components is implemented in the application.

Primary Service Handheld Business Components

The primary business components in the Siebel Service Handheld application are Action, Account, Service Request, Order, and Asset.

Action Business Component

The primary business component in the Service Handheld application is the Action business component.

NOTE: The Service Handheld application uses the ActionSHCE business object unlike other Siebel applications that use the Action business object.

Activities are based on the Action business component and, therefore, the following views and screens are based on the Action business component:

- Activities Activities view
- Activities Details view
- Parent applets in each Activities screen
- New Activity view in the Activities screen

The Service Handheld application is based on activities and all other data relationships are based on the Action object. By default, the activities that are selected for download are determined by a combination of the repository configuration, a business component filter, and a business object Filter. The business component filter is defined to select activities that meet both of the following criteria:

- Activities for which the end user is a team member or owner, and
- Activities for which the activity status is not Declined, Cancelled, or Done

Therefore, the end user sees only his or her open activities—that is, activities that require some action. Generally, the end user does not see activities for other users or activities that do not require action. For example, a field repair activity may be assigned to a field technician, and this activity automatically gets downloaded. There may be instances when you want activities that are assigned to another user to get downloaded. For example, a service center technician makes a follow-up call to the customer related to the activity assigned to the field technician. In order to download the follow-up call activity, you need to make the service center technician and the field technician part of the same team and assign the team to the activity.

This set of user's activities is further reduced by the default business object filters which specify a time frame for the activities:

- Today
- Past Due and Today
- Past Due till Tomorrow
- Today and Tomorrow
- Past Due, Today, and Tomorrow

The activities that are filtered for a user determine which associated accounts, service requests, and orders are downloaded. The association between the remaining business objects is configured in Siebel Tools.

Accounts Business Component

The business component filters and business object filters determine which activities (and their associated accounts) are downloaded. You can configure this relationship in Siebel Tools at the business object level by linking Account ID on Activity with ID on Account.

Service Requests Business Component

The downloaded accounts determine which service requests are downloaded. All service requests that are associated with the selected accounts are downloaded. This includes service requests that are associated with the downloaded activities, but it also includes all other service requests associated with the downloaded account. The relationship between accounts and service requests is configured in Siebel Tools at the component level by linking the Account ID on Service Request with the Account ID on Account. The service requests are further restricted by business component filters that select only service requests that are open or that were closed in the last 30 days.

Table 18 shows an example of the relationship between activities, accounts, and service requests.

Table 18. Activities, Accounts and Service Requests Relationship

Activity	Account	Service Request
1	A	X
	A	Y
2	A	X
	A	Z
	B	

Assume that Activity 1 is downloaded. Activity 1 is associated with one account (Account A) and one service request (Service Request X). Therefore, Activity 1, Account A, and Service Request X are downloaded. However, Account 1 has other service requests associated with it, Service Requests Y and Z, and these service requests are also downloaded.

In summary, the following records are downloaded:

- Activity 1
- Account A
- Service Requests X and Y

Activity 2 is not downloaded, although the account (A) and service request (X) associated with Activity 2 is downloaded. Account B, which is not associated with Activity 1, is also excluded. These exclusions illustrate the way in which the Action business component determines which of the other components are selected.

Orders Business Component

The downloaded service requests determine which orders are downloaded. Therefore, only those orders associated with the selected service orders are downloaded. Table 19 shows an example of the relationship between activities, accounts, service requests, and orders.

Table 19. Activity, Service Request and Order Relationship

Activity	Service Request	Order
1	X	L
2	X	M
	Y	N

Service Request X is associated with Activity 1. Any orders associated with Service Request X are downloaded. Therefore, in the example above:

- Order L which is associated with Service Request X is downloaded.
- Although order M is not associated with the downloaded activity (Activity 1), it is associated with the downloaded service request (Service Request X) and, therefore, it is also downloaded.
- Order N is not associated with Service Request X and, therefore, it is *not* downloaded.

Note that Activity 2 is not selected even though its associated service request (X) and order (M) are selected.

Assets Business Component

Accounts determine which assets are downloaded. Therefore those assets that are associated with the selected accounts are extracted. In addition, any assets in the field technician's trunk are downloaded.

No Record Business Object Filters

By default, all records for each screen in the Service application—Activities, Accounts, Contacts, and Service Requests—are downloaded to the device. If the defaults are applied, this would result in an unacceptably large number of records on the device. In the Service Handheld application, the relationship between the business components, as defined in Siebel Tools, determines which accounts, contacts, and service requests are selected and downloaded. Therefore, business object filters have been defined so that no account, contact, and service request records are selected for synchronization through the default mechanism. These no-record filters are:

- Def_*No Accounts
- Def_*No Contacts
- Def_*No Service Requests
- Def_*No Assets

These filters appear, by default, in the Choose drop-down list when end users synchronize their devices. (The exception to this is the Def_*No Contacts filter because there is no Contacts screen.) These filters are only applicable during synchronization. The asterisk in the filter name excludes them from the application's Queries drop-down list.

Child Service Business Components

Child business components of Activities. The Action business component is associated with the following child components, which are synchronized with the user's device:

- Expense Item
- FS Instruction
- FS Invoice
- FS Activity Parts Movement
- FS Activity Recommended Parts & Tools
- FS Activity Step
- Time Sheet Daily Hours

Child business components of Accounts. The following child components of the selected Accounts business component are downloaded:

- Asset Mgmt - Asset
- Contact Note
- Contact
- Entitlement Amount

Any warranties associated with the downloaded assets are downloaded.

Supporting Service Business Components

By default, all products, price lists, and rate lists for an organization are downloaded to the device. Since, for most businesses, this would result in a very large number of records being downloaded, default business component filters are provided that reduce the amount of data.

The preconfigured business component filters filter the data for products, price lists, and rate lists in the following way:

- **Products.** All products associated with an organization that meet the following conditions are downloaded: Orderable = TRUE and either Field Replaceable = TRUE or Bill Replaceable = TRUE.
- **Price Lists.** Price Lists have a valid time period as defined by the Start Date and End Date fields. All valid price lists for a user's organization are downloaded.
- **Rate Lists.** Rate lists have a valid time period as defined by the Start Date and End Date fields. All valid rate lists associated with the user's organization are downloaded.

Use these defaults as examples of how you can restrict the data in your organization. Identify a field that makes sense in your organization to use to restrict the data, and create a business component filter that specifies how data is to be extracted on this field. The following are other alternative methods for restricting data from these supporting business components:

- Increase the level of granularity of your organization. For example, you could break one large organization into several smaller organizations and assign different products, price lists, and rates lists to different organizations.
- Design multiple business component filters. This requires multiple servers, one for each business component filter.
- Add a field to each of the business components and filter on that field. For example, you could add a field called Product Family and use this field to categorize your products. Or you could add a Service Team field and use this field to classify your rate lists and price lists.

Parts is the only business component that is based on a user property. All parts that occur in the user's inventory are downloaded.

Entitlements are another supporting business component. The downloaded accounts determine the entitlements that are downloaded.

Summary of Service Handheld Filters

Table 20 shows the list of Service Handheld filters and the query defined for each filter. All filters are public—that is, they are visible to end users. Administrators can modify these business object filters in the Administration-Mobile > User Administration views in the Web client.

The Service Handheld application includes all business object filters for all of the Siebel Handheld applications, some of which are not relevant to Service Handheld. Therefore, it is recommended that you delete any Handheld business object filters that are not listed in Table 20 so that your end users do not attempt to use business object filters that do not apply.

Table 20. Service Handheld Filters

Business Object	Name	Query
Account	Def_*No Accounts	'Account'.Search = "[Id] IS NULL" 'Asset Mgmt - Asset'.Search = "[Id] IS NULL" 'FS Asset Warranty'.Search = "[Id] IS NULL" "'Entitlement Account'.Search = "[Id] IS NULL""
ActionSHCE	Def_Today's Activities	'Action'.Search = "[Planned] = Today() OR ([Planned] = """" AND [Created] = Today() AND [Created By] = LoginId())" "'Action'.Sort = "Planned""
Contact	Def_*No Contacts	'Contact'.Search = [Id] IS NULL""
SHCE Asset Management	Def_*No Assets	'FS Asset Measurement Characteristics'.Search = [Id] IS NULL""

Table 20. Service Handheld Filters

Business Object	Name	Query
Service Request	Def_*No Service Requests	'Service Request'.Search = "[Id] IS NULL" 'FS Invoice'.Search = "[Id] IS NULL" 'FS Invoice Line Items'.Search = "[Id] IS NULL" "'Order Entry - Line Items'.Search = "[Id] IS NULL""
ActionSHCE	Def_Today's Activities	'Action'.Search = "[Planned] = Today() OR ([Planned] = """" AND [Created] = Today() AND [Created By] = LoginId())" "'Action'.Sort = "Planned""
ActionSHCE	Past Due and Today	'Action'.Search = "[Planned] <= Today() OR ([Planned] = """" AND [Created] = Today() AND [Created By] = LoginId())" "'Action'.Sort = "Planned""
ActionSHCE	Past Due till Tomorrow	'Action'.Search = "[Planned] <= Today() + 1 OR ([Planned] = """" AND [Created] = Today() AND [Created By] = LoginId())" "'Action'.Sort = "Planned""
ActionSHCE	Today Due and Tomorrow	'Action'.Search = "[Planned] = Today() OR [Planned] = Today()+1 OR ([Planned] = """" AND [Created] = Today() AND [Created By] = LoginId())" "'Action'.Sort = "Planned""
Account	All_All Accounts	'Account'.Search = "" "'Account'.Sort = "Name, Location""
ActionSHCE	All_All Activities	'Action'.Search = "" "'Action'.Sort = "Planned""
Contact	All_All Contacts	'Contact'.Search = "" "'Contact'.Sort = "Last Name, First Name""

6 Deployment of Siebel Handheld

This chapter includes the following topics:

Overview of Server Installation in Support of the Siebel Handheld Client

This section provides an overview of important factors to consider when planning Direct Server Synchronization (DSS) infrastructure. One aspect of successful planning and execution of a Siebel Handheld DSS deployment is to understand hardware and network requirements. While these factors are generally applicable to all deployments, there are unique aspects to each customer configuration and implementation strategy. Therefore, it is highly recommended that customers conduct detailed configuration, sizing, and production readiness reviews with Siebel Expert Services. This is especially important for complex deployments, such as those supporting large numbers of users over wide geographic areas.

You can also improve the performance and scalability of DSS deployments by working with the Siebel application server component parameters. For more information about the Siebel application server, see "Optimizing Server Process Management for Direct Server Sync" on page 87.

NOTE: The recommendations in this chapter are general ones and are intended to raise your awareness of the key factors in handheld synchronization, performance, and scalability. These factors must be confirmed for your actual environment, and the usage patterns in your enterprise should be taken into consideration.

Server Topology Overview

The diagram in Figure 1 shows the servers and components that are required to deploy Direct Server Sync. The components are the Siebel database, Siebel server, Siebel Web engine, and Siebel Handheld. The synchronization components are the Siebel Sync Engine which resides on Siebel server and the Sync client on Siebel Handheld.

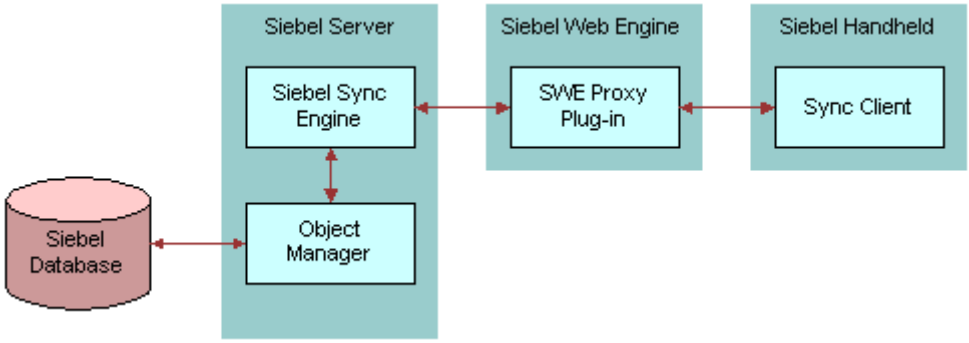

Figure 1. DSS Servers and Components

Recommended Hardware and Network Configuration

There are many factors to take into account when designing and implementing your server and network hardware and architecture, and sizing your database. The critical factor in sizing a handheld implementation is the expected peak synchronization load. Specific application variables to consider in calculating this load are: the number of concurrent users, number of transactions, and size of the data file. These variables should be considered within the context of the hardware specifications, including memory and processor speed.

The following resources are available as you plan your Siebel Handheld implementation:

- Refer to Microsoft and manufacturer documentation for comprehensive information on server hardware and registry parameters.
- Refer to Siebel Technical Note 405 for specific recommendations on the following topics:
  - Server Architecture
  - Server Hardware and Registry Parameters
  - Database Server Sizing
  - Database Tuning
  - Dial-Up and Network
  - Wireless
  - VPN (Virtual Private Networks)
- The Expert Services Handheld Review Package assists customers with planning their hardware and network for handheld implementations. Contact Siebel Expert Services for more information about this program.

Process of Server Installation for Siebel Handheld Direct Server Sync

The server components that are required for Siebel Handheld are installed when you install the Siebel server. See the Siebel Server installation guide for your operating system for details on installing and configuring Siebel server. The Siebel server installer copies the necessary DLLs to create an Object Manager for Direct Server Sync and creates and configures a Server Component and Component Group. The rest of this subsection describes which options you must select to correctly install the software, along with creating or configuring additional Object Managers.

This process includes the following tasks:

- "Installing the Direct Server Sync Components Using the Siebel Server Installer" on page 85
- "Configuring Direct Server Sync Logging Levels" on page 86
- "Optimizing Server Process Management for Direct Server Sync" on page 87

Installing the Direct Server Sync Components Using the Siebel Server Installer

Install the Siebel server, following the instructions in the Siebel server installation guide for the operating system you are using. When you reach the Setup Type screen in the Installer wizard, you are presented with three installation options:

- **Typical.** This setup option installs all Siebel server components except those displayed.
- **Compact.** This setup option installs only those components necessary to run the Siebel server, but no additional components or help.
- **Custom.** This setup option lets you customize your installation by choosing the specific components you want to install.

To install the Direct Server Sync components with your Siebel Server installation

1. During the Siebel Server installation, when the Setup Type screen appears, choose the Custom setting, and then click Next.

   NOTE: At this screen, you may also choose a different directory for your Siebel server installation.

2. From the following components, choose Handheld Synchronization:
   - Object Manager Component
   - Handheld Synchronization
   - Siebel Data Quality Connector
   - Remote Search Support
   - Siebel Java Integrator

   NOTE: You must enable Handheld Synchronization to enable Direct Server Sync for your DSS installation.

3 Click Next.

Continue with the installation in *Siebel Installation Guide for Microsoft Windows: Servers, Mobile Web Clients, Tools*.

The install wizard automatically installs and configures the DSS components.

DSS Server Component Parameters

See Table 21 for a list of required parameters. For information on adding DSS server components, see *Siebel System Administration Guide*.

Table 21. DSS Server Component Parameters

Parameter	Default Value	Description
ApplicationName	SiebelService for CE	The Siebel Handheld application. This value must match the name of the application as it is specified in Siebel Tools.
ApplicationTitle	Siebel Service	The name of the Siebel Handheld application as shown on the title bar on the browser. This parameter is only used if the Siebel application developed for the handheld is run on the Web client.
ApplicationSplashText	Siebel Service for CE	The Siebel Handheld application name as shown on the splash screen. This parameter is only used if the Siebel Tools application developed for the handheld is run on the Web client.
ComponentName	Siebel Field Service Client	Required for Direct Server Sync, it is entered in the registry on the handheld device.

Configuring Direct Server Sync Logging Levels

To set the logging level, you need to perform the following two tasks:

- "To configure server logging levels" on page 87
- "To configure server logging levels using the Server Configuration screen" on page 87

You set the logging level to a value between 1 and 5 as shown in Table 22.

Table 22. Direct Server Sync Logging Levels

Logging Level	Results
1	Error listing
2	Warning listing

Table 22. Direct Server Sync Logging Levels

Logging Level	Results
3	Information listing, including SQL tracing
4	Information listing
5	Debug

To configure server logging levels

1. Log into Server Manager from a (m:\siebel\bin\w32ud) prompt.
2. Run the following:

```
change evtloglvl HandheldSyncLog=[LogLevel] for comp [OMName], where [LogLevel] = an
integer between 1 - 5 (5 enables all logging) and [OMName] = the OM name (for example,
<ApplicationName>CEObjMgr_enu).
```

To configure server logging levels using the Server Configuration screen

1. Navigate to Administration - Server Configuration > Servers > Components.
2. In the Component list, query for the appropriate component, such as Handheld Sales CE (ENU).
3. In the Events list, search for the event type Handheld Sync Logging.
4. Set the value to a number between 1-5 and then save the change.

   See Table 22 for logging level results.

Optimizing Server Process Management for Direct Server Sync

Several parameters are outlined below that help optimize server processing for Direct Server Sync.

To set parameters to optimize server process management

1. Log into the Siebel client as a user with Administrative responsibility.
2. Navigate to Site Map > Administration - Server Configuration > Servers > Components > Parameters.
3. In the Component Parameters sub-view, query for SalesCEObjManager.

4 Use the following table to update the parameters.

Parameter	Description
Max Tasks	Determines the maximum number of threads that can run concurrently for a given component. You should set the value of this parameter to a number equal to that of the expected peak number of concurrent synchronizations.
Min MT Servers	Determines how many multithreaded server processes are started by default for the Siebel Object Manager.
Max MT Servers	Defines the maximum number of processes supported by the instance of the Siebel Object Manager. The value for this parameter should be the same as Min MT servers and should be set so that threads per process (Max Tasks/Max MT Servers) is optimized.

About Siebel Handheld Synchronization Performance and Scalability

There are a number of factors that affect the performance and scalability of handheld synchronization. It is very important that administrators and those involved in the design and deployment of Siebel Handheld understand the impact of usage parameters on synchronization performance and scalability. For more information, see Technical Note 405 for application-specific variables affecting synchronization and data from synchronization scalability tests.

Using Multiple Synchronization Servers

As you scale your deployment, you may need to use multiple synchronization servers to accommodate all of your users. Your end users must be manually assigned to a particular server. Resonate products cannot be used for load balancing. Refer to Technical Note 405 and to your hardware manufacturer's documentation for more information on scaling deployments.

Providing Synchronization Security with Secure Sockets Layer

The synchronization data stream can be secured using Secure Sockets Layer (SSL). To employ SSL security:

- Make sure that a certificate has been created and installed on all Web servers that service handheld synchronization.

Use the following steps to install the root certificate on each handheld device.

To install the root certificate on the handheld device running PPC 2003 or newer

1. Navigate to the certificate authority server from your desktop computer.
2. Select the first option (retrieve the CA certificate or certificate revocation list), and click Next.
3. Click on Download CA certificate.
4. Click Save, and save as certca.cer on your desktop.
5. Use ActiveSync to copy the downloaded certificate, certca.cer, from the desktop to the My Documents folder on the handheld device.

   NOTE: Make sure the certca.cer file is copied into the My Documents folder.

6. Go to My Documents and tap the certificate icon.

   The certificate should be installed. You can validate it by going to Settings > System > Certificate.

7. Click the Root tab.

   You should see the new certificate on the list.

8. Open up a browser and run the following command, substituting the appropriate server name, application name, UserName and Password (in red below).

   https://<SERVERNAME>/<APPLICATION NAME>/start.swe?SWEExtSource=siebelpda&SWEExtCmd=ExecuteLogin&UserName=<USER LOGIN>&Password=<PASSWORD>

9. When a dialog appears, click Yes.

To complete root certificate installation on PPC 2002 device

1. From a web browser, navigate to the Microsoft Web site and download and launch the following executable file:

   AddRootCert.exe.

2. Double-click to open the executable file.

   A list of certificates appears.

3. Select certca.cer.

   Information for the certca.cer Certificate appears.

4. Click Install Certificate.
5. Following successful installation, click OK.

Process of Installing the Siebel Handheld Application

Before you deploy the Siebel Handheld application, there are several tasks you need to perform.

NOTE: If you are setting up your Siebel Handheld application to allow end user translation functionality, see "Installing and Setting Up End User Translation Functionality" on page 32.

This process includes the following tasks:

- "Editing the Handheld Application Installation Configuration File" on page 90
- "Enabling Handheld User Database Backup" on page 93
- "Deploying Patches to the Handheld Application with Direct Server Sync" on page 96
- "Downloading Patches to the Handheld Device" on page 97

Editing the Handheld Application Installation Configuration File

The Siebel Handheld application setup.ini file configures the Siebel Handheld Client application installer. Before you deploy a custom-configured Siebel Handheld application, you must update the parameters in the setup.ini file so that the correct values are written to the device registry before distributing the client installer to end users.

The setup.ini file resides in the Siebel Handheld application directory. To modify this file, create a Siebel Handheld installation directory on the network, or create and duplicate a custom DVD-ROM for distribution to end users. This installation directory must include the same files and directory structure as the original DVD-ROM. Then, modify the setup.ini file in the custom installation directory.

NOTE: If you wish to set up parameter values other than default values, add the parameter to the setup.ini file and add the proper values.

Table 23 lists the parameters that you may need to modify in the setup.ini file, their default value, and a description of the parameter.

Table 23. Siebel Handheld setup.ini Parameters

Parameter	Default Value	Description
Version	7.8	Version of the desktop Siebel application against which the Siebel Handheld version synchronizes.
Application Title	SiebelService for CE	The Siebel Handheld application. This value must match the name of the application as it is specified in Siebel Tools.
ApplicationName	SiebelService for CE	The Siebel Handheld application. This value must match the name of the application as it is specified in Siebel Tools.

Table 23. Siebel Handheld setup.ini Parameters

Parameter	Default Value	Description
ApplicationSplashText	Service Handheld	The Siebel Handheld application name as shown on the splash screen.
EnableRemote	N	Used for Integrated Sync. The default value is "N". Valid values are "Y" = [Enable Remote Sync] checkbox is selected, "N" = [Enable Remote Sync] checkbox is unselected in handheld sync window.
MultiCurrency	True	Required for the application to handle multiple currencies.
SyncURL	http://*Web Server Name or IP address/ Virtual Directory Name*	Required for all synchronization methods. This value is entered in the registry on the handheld. See "Changing the SyncURL on Siebel Handheld Devices" on page 95 for more information on this parameter.
[EndUserLocalization] Section		
User DefinedLang	Y	
LogUserDefinedLangErrors	N	
[Printing] Section		See Appendix F, "Print Configuration Settings," for more information on the parameters used to configure printing.
[Backup] Section		Enables handheld database backups. See "Enabling Database Backup" on page 94 for more information on settings for the parameters in this section of the setup.ini file.
[ApplicationLocation] Section		Specifies where the Siebel Handheld application and database are installed. See "Setting Up Siebel Handheld Application Installation on a CompactFlash Card" on page 94 for more information on the parameters in this section of the setup.ini file.
InternetOptionReceiveTimeout	600000	Length of time to receive data (except data extraction) from the server before the handheld client times out. This parameter only applies to synchronizations using Direct Server Sync. In milliseconds. The default value, 600000, is 10 minutes.

Table 23. Siebel Handheld setup.ini Parameters

Parameter	Default Value	Description
InternetOptionSendTimeout	600000	Length of time to send data to the server in a single call the handheld client times out. This parameter only applies to synchronizations using Direct Server Sync. In milliseconds. The default value, 600000, is 10 minutes.
InternetOptionDataExtractionTimeout	900000	Length of time to extract data from the server to the handheld device the handheld client times out. This parameter only applies to synchronizations using Direct Server Sync.
[Logging] Section		
TXNErrorLevel	1	This level controls the extent of transaction error information received from the server. The default value is 1. Valid values are 0 = no log information, 1 = report txn errors in the client log, txnerror.hh. A "Transaction Errors Reported" message appears during sync when txn errors occur. To view errors, double click the message.
DefaultLogLevel	2	The default value is 2. You can set the value to any of the following: -1 = NO LOGGING 0 = LOG_SYSTEM 1 = LOG_ERROR 2 = LOG_WARNING 3 = LOG_INFO, INCLUDING SQL TRACING 4 = LOG_DEBUG The log messages with level equal to or lower than the set value will be logged. For example, if the default log level is set to 2, the messages belonging to System, Error, and Warning log levels will be logged. If the DefaultLogLevel is -1, no logging will be provided.
MaxLogFileSize	2000	This is the maximum log file size in KB. If the program finds the size of the latest log file is greater than this value, a new log file should be used for logging.

Table 23. Siebel Handheld setup.ini Parameters

Parameter	Default Value	Description
MaxSavedLogFiles	3	Determines the number of log files to save on the device. It does not include the one currently used by the program. If the value is set to 2, then the valid log file names are siebel_log.txt, Siebel_log1.txt, and siebel_log2.txt. Siebel_log.txt is the current log file. Siebel_log1 is an older one and Siebel_log2 is the oldest.
PressureThreshold	29	Prompts user to restart application after 30 visits, thereby freeing application memory.

Enabling Handheld User Database Backup

Siebel Handheld infrastructure provides a way to automatically back up a snapshot of the user's database and current settings, and to restore those settings in the event of a total device failure. The backup functionality automatically copies all the necessary files to a single compressed file. Each time the user backs up, the backup process creates a new copy of the file and deletes older backups. The backup process must close the application in order to copy the RDBMS and handheld configuration files. After the files are copied, it automatically restarts the application.

The backup feature must be enabled in the application. If the backup feature is enabled, a backup is created during every synchronization. This way, the user retains an up-to-date snapshot of the database, and older backup versions are removed.

When the database backup feature is enabled, a database backup occurs automatically at the end of each synchronization. This makes the overall time to synchronize somewhat longer. The size of the database and the speed of the CompactFlash card also affect the time it takes to complete synchronization.

NOTE: Backups should not be used as a replacement for synchronization.

Setting Up Database Backups

In order to back up the database you must do the following:

- Provide users with an external memory card.
- Install the Siebel Handheld application with the backup functionality enabled. See, "Enabling Database Backup" on page 94.
- Instruct users to back up their database on a regular basis.

Backing Up to External Media

It is recommended that you back up to external media—for example, a CompactFlash card or the embedded backup storage that is included in many devices.

Enabling Database Backup

Before users can back up the database on their handheld devices, you must first enable the Backup menu item on the Siebel Handheld application. This requires editing the setup.ini file to enable backup.

NOTE: Backup should be kept on external media or built-in storage. Trying to set the location to the main memory, such as \My Documents, will not work.

The following is an example of the default settings for the backup parameters:

```
[Backup]
BackupLocation = \Storage Card\Siebel Backup
BackupEnabled  = N
```

- BackupEnabled = [Y/N] The default is N (No). Change this setting to Y (Yes) to enable the Backup menu item and to trigger a backup to occur automatically after each synchronization.

  If the product is installed with Backup = N, the backup menu item is disabled. You need to change the parameter, then reinstall the product in order to enable the menu item.

- BackupLocation = [Backup location] The default is an external storage card. Specify the location where you want the data backed up.

For more information on the setup.ini file, see Chapter 8, "Installing the Handheld Application."

Restoring from a Backup

For information on restoring a database from a backup, see "Backing Up and Restoring Data with the Siebel Handheld Client" on page 119.

Installing Print Templates in the Siebel Handheld Install Directory

If your application allows printing, you must include all print templates in the template subdirectory of the Siebel Handheld Install directory. The print templates are language-specific, so copy the templates to the template subdirectory within the language directory of the language you are deploying (for example, ...\ENU\Template). If the template includes double byte characters, such as Japanese string, you must save the template files as Unicode (UTF16) files. For supported language codes, see *System Requirements and Supported Platforms* on Siebel SupportWeb.

See Appendix F, "Print Configuration Settings" for more information about Printing.

Setting Up Siebel Handheld Application Installation on a CompactFlash Card

The default, when installing Siebel Handheld applications, is to install both the application and the database into RAM on the device. In most instances, there is sufficient available memory, and end users are satisfied with the application performance. There are situations, however, where you may need to optimize storage on the device.

If you need to maximize main memory availability, you can install the application binary files on an external location such as a CompactFlash card. Installing the data on the CompactFlash card is not recommended because performance likely degrades.

You can specify, in the application's setup.ini file, where the application files and database files are to be installed. By default, this is set to the handheld device. In the ApplicationLocation section of the setup.ini file, there are two parameters. The following shows the default settings for these parameters:

```
[ApplicationLocation]
ApplicationBinariesLocation  =  Device
ApplicationDataLocation      =  Device
```

- ApplicationBinariesLocation = Device. The default is Device. The other value is CF (CompactFlash).
- ApplicationDataLocation = Device. The default is Device. It is recommended that you always install the data files on the device.

After you have edited the setup.ini file, follow the instructions for installing the Siebel Handheld application on the client.

Changing the SyncURL on Siebel Handheld Devices

For Direct Server Sync (DSS), SyncURL is used to locate the Web Server and the virtual directory so that the handheld synchronization client can communicate with the handheld synchronization server. For example:

SyncURL = http://web server name or IP address/virtual directory name

You can reset the SyncURL by re-installing the application with the updated setup.ini file.

Installing the Siebel Handheld Application from External Media

Users who synchronize in DSS mode may not have a PC, and so may not be able to create a partnership and synchronize using Microsoft ActiveSync. You can install the application for each user through a standard partnership, or you can configure a stand-alone installer.

During the installation, the InstallShield Wizard screen prompts you by asking,

```
Would you like to deploy a CAB file to enable rapid installation for additional
devices?
```

You can configure a stand-alone installer by selecting Yes.

You are prompted to select a language for the application and to specify a location for the CAB file. The installer creates the CAB file in the location you specify.

The CAB file is a self-extracting file that includes all the files needed to install the Siebel Handheld application on the handheld device. When you single-click the file, the CAB file installs the application.

You may create an image of the Handheld Installer that can be loaded onto external media such as a CompactFlash card. When you create a stand-alone installer platform, a Cabinet file (CAB) is created in a location that you specify. The CAB file name should be siebel.pocketpc_arm.CAB. You can copy the CAB file to a CompactFlash card, insert the card into the handheld device, and single-click on the file to install the Siebel Handheld application on the device.

NOTE: When installing the application on the device, the end user is asked if he or she wants to install the application in the default directory. Regardless of the user's answer, the application is always installed in the directory specified by the setup.ini file. Therefore, the only way to change the directory the application is installed in is to modify the parameters in the ApplicationLocation section of the setup.ini file.

Distributing Siebel Handheld Application Patches

The Siebel Handheld patch delivery mechanism, PatchAgent, is a tool that automates the deployment of patches to handheld devices as part of the synchronization process. It allows administrators to distribute patches to end users with minimal involvement by the users.

During synchronization the PatchAgent executable is called, and if it detects a handheld patch, it automatically installs the patch on the handheld device. In order for PatchAgent to detect and deploy the patch, the patch files must be created and placed in the directory where PatchAgent expects to find the files.

Deploying Patches to the Handheld Application with Direct Server Sync

In order to set up patch deployment for Direct Server Sync users, you need to patch the Siebel server and patch one handheld client which serves as the "model" from which patch files are pulled. Then the patch files are then packaged and placed in the location on the server where PatchAgent expects to find the files. The next time the end user synchronizes with Siebel server, the patch is installed onto his or her device.

For more information on how the patch process works with synchronization, see "Downloading Patches to the Handheld Device" on page 97.

To set up patch deployment

1. On Siebel server, verify that the following directories exist. If they do not, create the directories.
   - C:\patch\AppBinDir

- C:\patch\WinDir

NOTE: You may specify a drive other than C:. However, the directory names must be specified as shown above.

2. Apply the patch to Siebel server.
3. Apply the patch to one handheld client.
4. Examine the handheld directory (My Device\Program Files\Siebel Handheld) and copy the files with the latest date to the Siebel Server directory (C:\patch\AppBinDir).
5. Examine the handheld directory (My Device\Windows) and copy the files with the latest timestamp to the Siebel Server directory (C:\patch\WinDir).
6. From the DOS prompt, change directory to *Siebel Root*\SWEApp\BIN.
7. Execute the following command to stage the patch files and the patchlist.txt in the correct location on the server:

   C:*Siebel Root*\SWEApp\BIN>createpatchlist /s c:\patch /t c:*Siebel Root*\sweapp\cepatch

   /s – source directory. The directory where the patch files were moved.

   /t – target directory. The directory where PatchAgent checks for the patch files.

   NOTE: You may specify a drive other than C:. However, the location of the source and target directories must be specified exactly as shown.

   After the patch files have been moved and the patch list created, the following message appears: "Patch List has been successfully created in *Siebel Root*\SWEApp\cepatch files transferred successfully."

The patch files are now ready to be deployed to Direct Server Sync end users.

Downloading Patches to the Handheld Device

Patches are downloaded as part of the synchronization process. Once the patch is staged, the patch is downloaded the next time the end user synchronizes the handheld with Siebel server. The end user synchronizes as he normally would using Direct Server Sync. First, any transactions are uploaded from the handheld device. Then:

- If a patch needs to be applied to the handheld, the synchronization process terminates, and the patch process is started, which downloads the patch.
- If no patch needs to be applied, the synchronization process continues until the process is complete.

After a successful patch installation, Siebel Handheld Sync will launch automatically and continue the extraction.

If the patch files are not successfully downloaded, the end user is prompted to restart the patch process. The end user must locate and launch patchagent.exe manually. Then, once the patch is successfully installed, the end user is prompted to complete the synchronization.

7 Synchronization Conflict Handling and Recovery with Handheld

This chapter includes the following topics:

Preventing Synchronization Transaction Conflicts with Siebel Handheld

When multiple end users synchronize with a remote database, transaction conflicts may occur which can result in data loss. The Extended Pick and Insert Failure functionality minimizes synchronization conflicts. If transaction conflicts are unavoidable, then Siebel Handheld Sync Service captures the transaction information so that the data can be recovered. The combination of these features eliminates the potential for data loss.

There are two features that minimize synchronization transaction conflicts.

- Extended Pick processing
- Insert Failure processing

The next several sections describe these features in detail.

Extended Pick Processing in Siebel Handheld

The handheld is used to capture and record transactions entered by the handheld user. During synchronization, these transactions are processed on the server. A situation may arise in which a transaction fails when it is applied to the Siebel server (through Direct Server Sync). For example, a Siebel Handheld user creates a professional call activity for one of his contacts. However, since he last synchronized with the server, the contact involved in this transaction was deleted from the server database. When he tries to synchronize his handheld, the server does not find his contact, and the call activity for the contact cannot be inserted into the server database. To generalize this example, any transaction that involves data selected through picklists or pick applets fails on the server if that pick data is not present at the time of synchronization.

With many handheld users synchronizing with Siebel server and modifying data at different times, this scenario is not uncommon.

The following describes how a pick transaction is recorded on the handheld device and processed on the server during synchronization. Assume that the handheld user wants to create a new call activity for the contact. In order to do this, the user inserts a new record in the Professional Call screen and "picks" the desired contact. When the contact is selected on the handheld, the Row Id of the contact (Contact Id) is recorded. However, on the handheld, the application does not record any of the pick map fields associated with the contact; only the Contact Id is recorded. When the user synchronizes his handheld, that Contact Id is used to locate the contact record in the server database. The Contact Id is then used to retrieve all of the fields used in the pick map for the Contact Id (Last Name, First Name, Middle Initial, Address, City, State, Zip Code, and so on).

In order to minimize pick processing failures, the handheld must record the pick row Id and all of the pick map data that is required for a successful transaction. This data must be recorded at the time of the pick, so that it can be passed to the server for processing. This enhanced functionality is called Extended Pick processing. Now, when the user synchronizes, the transaction is processed on the server as follows:

- The synchronization process first looks for the pick record on the server database. For example, to add an activity it looks for the contact and, if it finds the contact, it retrieves the necessary data and adds the activity. This is the default behavior.
- If the server does not find the pick map record in the database and extended pick processing is enabled, it retrieves the pick record Id and pick map data from the handheld transaction and applies the data directly to the server.

Enabling Extended Pick Processing in Siebel Handheld

You can enable Extended Pick processing globally or at the business component level when configuring the Siebel Handheld application. When the *DefaultHandheldSyncPickMode* parameter is enabled every handheld pick transaction is subject to pick processing.

To enable extended pick processing globally

1. From the application-level menu, choose Navigate > Site Map > Administration-Mobile > Application Administration.

2 Click the Settings view tab, then Query on the following value:

DefaultHandheldSyncPickMode

If there is no such record, a new record with that value should be created. See Table 24 for the values to enter.

Table 24. Application Settings

Setting Name	Value	Description
DefaultHandheldSyncPickMode	NoFail	Turns Extended Pick processing ON.
	FailOnError	Turns Extended Pick processing OFF This is the default behavior if the parameter is not specified in the Settings view.

To enable extended pick processing at the business component level

- When enabling extended pick processing at the business component level, the user property is defined on specific business components. The syntax for setting Extended Pick on a business component and the parameter values (Table 25) follow:

  HandheldSyncPickMode=*Fieldname*|*Mode*

Table 25. Business Component User Property Parameters

Business Component User Property	Parameter	Value	Description
HandheldSyncPickMode	*Fieldname*	—	Specify the field name as defined in Siebel Tools.
	Mode	NoFail	Turns Extended Pick processing ON.
		FailOnError	Turns Extended Pick processing OFF.

You may apply Extended Pick processing to multiple fields in a single business component, using the following syntax:

HandheldSyncPickMode=*Fieldname1*|*Mode1, Fieldname2*|*Mode2*

The setting on the business component takes precedence over the setting defined in the Settings view. For example, you can use the business component setting to nullify the global application of Extended Pick. Conversely, you can enable Extended Pick processing on specific business components if the setting in the Settings view has disabled the functionality.

For more information, see *Using Siebel Tools*.

Table 26 shows the combined effect of the settings in the Application-Mobile > Settings View and on a business component.

- The DefaultHandheldSyncPickMode setting is omitted from application settings.

- If there is no setting on a particular business component, the setting defined in the Settings view applies.
- If there is no setting defined in the Settings view, or on the business component, the default behavior is for Extended Pick to be turned OFF.

In Table 26, read down the Tools configuration column and across to the Mobile Administration View Setting to see the combined result. For example, a NoFail configuration in Tools combined with a FailOnError setting in the Mobile Administration View results in the combined effect NoFail.

Table 26. Combined Effect of Extended Pick Settings for Business Components

Tools Configuration (User Property) **HandheldSyncPickMode=**	**Mobile Administration View Settings** **DefaultHandheldSyncPickMode=**		
	NoFail	**FailOnError**	**Omitted**
NoFail	NoFail	NoFail	NoFail
FailOnError	FailOnError	FailOnError	FailOnError
Omitted	NoFail	FailOnError	FailOnError

Insert Failure Processing in Siebel Handheld

When the server executes an insert transaction and the insert fails, there are two ways the insert failure can be handled:

- A record is logged that indicates the insert failed, and synchronization continues. This is the default insert behavior. The Conflict Administration screen can be used to recover the data.
- The insert failure is logged, including details of the failure, and synchronization is interrupted. This is the FailOnError behavior. FailOnErrore behavior guarantees that, should an insert fail, data loss does not occur since synchronization is stopped until the circumstances causing the failure are resolved.

NOTE: If FailOnError processing takes place, synchronization is interrupted. The synchronization process continues to fail until the problem is resolved. End users must contact their system administrators to recover from the failure. The recovery effort may involve modifying transactions queued for processing, modifying data on the server, or reentering transactions through the Siebel Handheld application.

Handling Insert Failures

You can enable Extended Pick processing globally or at the business component level when configuring the Siebel Handheld application. The following setting applies globally throughout the application. When FailOnError is specified, every failed insert interrupts synchronization.

DefaultHandheldInsertMode.

You can specify the value for this setting through the Settings view in the Administration-Mobile screen.

Table 27 shows the settings for this parameter.

Table 27. Insert Failure Processing Parameter Setting

Parameter	*Mode* Value	Description
DefaultHandheldInsertMode	NoFail	Insert failures are logged and synchronization continues. This is the default behavior.
	FailOnError	Insert failures are logged and synchronization terminates. The system administrator must intercede to recover from this failure.

Optionally, you can enable Extended Pick processing at the business component level.

HandheldInsertFailMode=*Mode*

The user property is defined on specific business components. The setting in the business component takes precedence over the setting defined in the Settings view. Table 28 shows the syntax for setting the Insert Fail Mode on a business component and the parameter values. These settings are made in Siebel Tools.

Table 28. Business Component User Property Parameter

Business Component User Property	Parameter	Value	Description
HandheldInsertFailMode	*Mode*	NoFail	Insert failures are logged and synchronization continues. This is the default behavior.
		FailOnError	Insert failures are logged and synchronization terminates. The system administrator must intercede to recover from this failure.

Table 29 shows the combined effect of the setting in the Settings view and the setting on a business component.

- The business component setting always takes precedence over the settings defined in the Administration-Mobile > Settings view.
- If there is no setting on a particular business component, the Applications Settings value applies.
- If there is no setting in either the Settings view or on the business component, the default behavior is the same as NoFail. Insert failures are logged and synchronization continues.

In Table 29, read down the Tools Configuration column and across to the application settings value to see the combined result. For example, a NoFail BC setting combined with a FailOnError application setting results in the combined effect NoFail.

Table 29. Combined Effect of Insert Failure Settings

Tools Configuration (User Property) **HandheldInsertFailMode=**	**Application Settings Value** **DefaultHandheldInsertMode=**		
	NoFail	**FailOnError**	**Omitted**
NoFail	NoFail	NoFail	NoFail
FailOnError	FailOnError	FailOnError	FailOnError
No Setting on Business Component	NoFail	FailOnError	NoFail

Recovering Error Data in Siebel Handheld

Selected synchronization and conflict errors are captured to the database, and can be accessed through the Conflict Administration screens. This mechanism is always enabled. For more information on recovering error data, see "Obtaining Error Details from Siebel Handheld Synchronization Sessions" on page 66.

8 Installing the Handheld Application

This chapter includes the following topics:

Preparing to Upgrade from a Previous Siebel Handheld Release

The Siebel Handheld application cannot be upgraded directly on the device. To upgrade your existing application and to migrate your data and settings to Siebel 7.8 use the following process:

- Back up the handheld device by synchronizing to your database. See, "Synchronizing Data on the Siebel Handheld Device" on page 120.
- Uninstall your Siebel Handheld application. See, "Uninstalling the Siebel Handheld Application" on page 107.
- Then install the new version of your Siebel Handheld application. See "Installing the Siebel Handheld Application" on page 105.

Installing the Siebel Handheld Application

There are two ways to install software onto any Windows-powered handheld device. The first is to create a partnership between a desktop or laptop PC and the handheld device using Microsoft ActiveSync. The second is to install the application from type of external media.

NOTE: It is important to remember that end users must synchronize their handheld devices to download the application configuration and populate the database before they can use the application. Therefore, an administrator may install the application for an end user, but the synchronization must occur using the end user's name and password.

Installing Handheld for Direct Server Sync Users

Most DSS users do not have access to a PC for creating a partnership with their handheld device. In this case, the administrator has to create a temporary partnership with each user's device and install the application. Siebel Handheld applications can be installed in DSS mode using the Siebel Installer as described in this section.

To install the Siebel Handheld application using DSS

1. Using Microsoft ActiveSync, establish a partnership between the handheld device and the PC.
2. On the PC network or DVD-ROM, locate the folder where the Siebel Handheld Application folder is installed, then double-click setup.exe.

   The installation process begins.
3. From the Choose Setup Language window, choose the language you want to use for the installation and click OK.

   The next screen displays and asks, "Would you like to deploy a CAB file to enable rapid installation for additional devices?"

   NOTE: If you want to create a stand-alone installer, choose Yes, and then click Next.
4. To install the application on the device, click Next.
5. Select the desired language of the Siebel Handheld Application, and click Next.

   A window appears showing the setup status, followed by several DOS windows.
6. The Add/Remove Programs dialog appears. This is followed by the Installing Applications dialog that asks if you want to install in the default directory. Click Yes to install the Siebel Handheld application on the handheld device.

   NOTE: The default directory in the Applications dialog is the directory specified by the system administrator in a setup file. So that the application works correctly, the Siebel Installer installs the application in the default directory. Therefore, even if you specify a different location for the application, the application is installed in the default location.

   The Siebel Handheld installation automatically installs SSCE Service Pack 1 and overwrites existing files on your system. A message may appear asking if you want to replace existing files. Respond by tapping "Yes to All."

   During the installation, the Installing Applications dialog box displays a progress bar. It may take several minutes to install on the handheld device. When the application has been downloaded to the handheld device, the Application Downloading Complete dialog box displays on the PC.
7. Click OK.

   A window appears informing you that setup has finished installing the Siebel Handheld application on your computer.
8. Click Finish.

After installing the Siebel Handheld application and Siebel Handheld Sync, you must perform an initial synchronization before the application can be used. Refer to "Synchronizing Data on the Siebel Handheld Device" on page 120 for the procedure for synchronizing your data.

See Chapter 5, "Data Filtering," for more information on using Direct Server Sync and setting filters.

Reinstalling the Siebel Handheld Application

Follow the steps below to reinstall your Siebel Handheld application.

To reinstall the Siebel Handheld application

1. Before you reinstall the current application, you must first uninstall the current application on your device. See "Uninstalling the Siebel Handheld Application" on page 107 for more information.
2. Once you have uninstalled the application, you may then reinstall the application or install the newer version. See "Installing the Siebel Handheld Application" on page 105, and follow the instructions for installing the Siebel Handheld application.
3. Synchronize your Siebel Handheld application. See "Synchronizing Data on the Siebel Handheld Device" on page 120 and follow the instructions for synchronizing your data.

Uninstalling the Siebel Handheld Application

Uninstalling the application deletes the database files as well as the application files.

CAUTION: Before you uninstall your application, you must synchronize your database to save any changes to your data since the last synchronization. If you do not synchronize before uninstalling, these changes will be lost. See "Synchronizing Data on the Siebel Handheld Device" on page 120 and follow the instructions for synchronizing your data.

To uninstall the Siebel Handheld application

1. Select Start > Settings.
2. Select the System tab and tap Remove Programs.
3. From Remove Programs, select the Siebel Handheld application and tap Remove.

   NOTE: A dialog box appears saying that a file cannot be deleted. This occurs because a file may still reside in memory. Ignore this message and tap OK to delete all application and database files. No additional steps are required to remove the application and database files.

9 Working with Siebel Handheld Applications

This chapter includes the following topics:

Components of the Siebel Handheld Interface

The components of the handheld interface include the Screens menu, Show drop-down list, application-level menu, Queries drop-down list, toolbar, and status bar.

Figure 2 shows the components of the Siebel Handheld application.

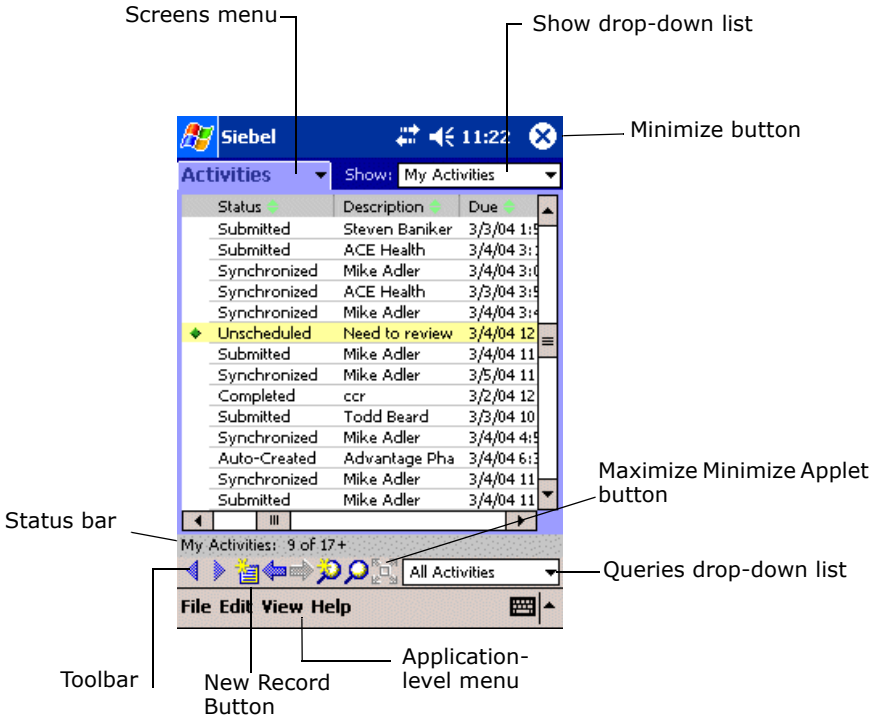

Figure 2. Siebel Handheld Screen Components

Screens Menu

The screens menu is the first level of navigation in the Siebel Handheld application. When you launch the Siebel Handheld application for the first time, this tab is labeled Screens unless you have configured a start up view to appear. When you tap the screens menu, a drop-down list of all available screens appears. Tap the desired screen in the list to navigate to it.

For information about configuring a start up view, see "Setting User Preferences in the Siebel Handheld Application" on page 118.

Show Drop-Down List

The Show drop-down list is the second level of navigation in the Siebel Handheld application. This is where you choose views for the selected screen.

Application-Level Menu

The application–level menu consists of the File, Edit, View, and Help menus. Tap a menu to select a menu option. The options that are available in each menu vary depending on the task you are performing.

Queries Drop-Down List

Up to two predefined queries, or business object filters are displayed in the Queries drop-down list. The list displays the query chosen when you synchronized (or the default filter if you did not change the Set Filters selection) and the All Records query, if it is defined. For more information, see Chapter 5, "Data Filtering."

Siebel Toolbar

The Siebel toolbar has buttons for frequently used functionality like record navigation and querying. You can configure your toolbar and specify which buttons are displayed. See "Configuring User Interface Elements in Siebel Handheld" on page 25.

- Tap the New Record button to add a new record to the active list or form.
- Tap the Back button to navigate to the previous view.
- Tap the Forward button to navigate to the view you were in before you selected the Back button.
- Tap the Record Navigation buttons to move through records in a form or a list.
- Tap the New Query button to begin a new query in a form or a list.
- Tap the Execute Query button to run the query you just created.
- Tap the Maximize Minimize Applet button to expand the active applet so that it is the only applet displayed and takes up the entire screen display. This allows you to see more columns or more rows of data at one time. Tap again and the screen displays two applets.

Customizing the Toolbar

You can customize which buttons appear on your toolbar by selecting View > Customize Toolbar from the application menu.

To add and remove buttons from the toolbar, drag and drop the desired buttons between the Choose Toolbar Buttons dialog box and the toolbar.

Minimize Button

The Minimize button appears in the upper-right corner of your window. When you tap the Minimize button, the application minimizes, but does not close. To reopen the minimized application, from the Start menu, tap the Siebel Handheld icon. To close (exit) the application, choose File > Exit.

Do not confuse the Minimize button that appears in the upper right corner with the Maximize Minimize Applet button that appears on the Toolbar.

Status Bar

The status bar appears at the bottom of the application window. It provides information about the current view.

Siebel Store-and-Forward Messaging and Asynchronous Query Icons

If your Siebel Handheld application is enabled with Siebel Store-and-Forward Messaging, three icons appear in the top-right corner of the screen. They are the Inbound Message icon, Asynchronous Query icon, and Outbound Message icon. Figure 3 displays the Store-and-Forward Messaging and Asynchronous Query icons. For more information on Siebel Store-and-Forward Messaging and Asynchronous Query, see *Siebel Store-and-Forward Messaging Guide*.

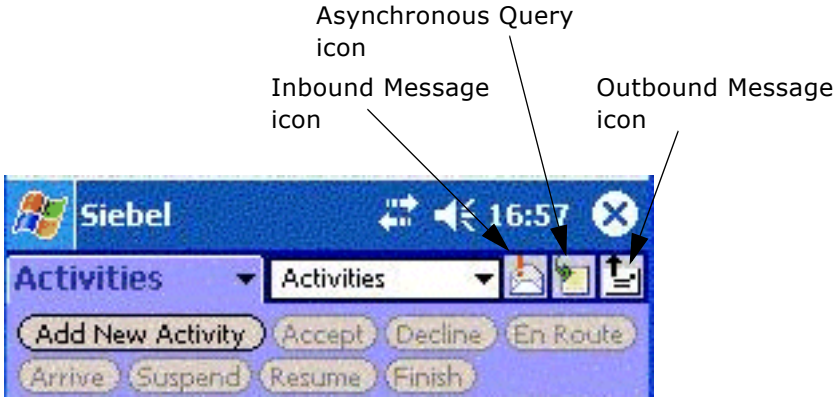

Figure 3. Details of Siebel Store-and-Forward Messaging and Asynchronous Query icons

Navigating the Siebel Handheld Interface

Navigating the Siebel Handheld application is achieved through the screens menu, the Show drop-down list, drilling down, drilling across, toolbar buttons, and toggling.

Screens Menu

The screens menu provides the first level of navigation. Here you select the screen you want to work in. For more information about the screens menu, see "Components of the Siebel Handheld Interface" on page 109.

Show Drop-Down List

The Show drop-down list is the second level of navigation. After you have selected a screen from the screens menu, select a view from the Show drop-down list. Views in the Show drop-down list are specific to each screen.

For example, in the Activities screen, you can pick the Recommended Parts & Tools view from the Show drop-down list.

Toggling Between Applets

A maximum of two applets can be displayed at one time in a Siebel Handheld application. In situations where there are more than two applets configured and a Toggle button has been configured, you may tap the Toggle button on the applet. Or you may select View > Toggle from the menu to toggle between applets.

Expandable Fields

Due to the limited screen size on the handheld, it is not always easy to view the data in a field without scrolling or resizing the field width. Therefore, some fields that could contain large amounts of text are expandable—that is, the data can be viewed in a text box. An ellipsis (...) appears in the field to indicate that the field is expandable. When you tap the ellipsis, a text box opens showing the data in that field. When you step off the field, the text box closes.

In form applets, an ellipsis is always visible in those fields that are expandable. However, in list applets, due to space constraints, the ellipsis is not always visible in expandable fields. When you tap on an expandable field in a list applet, the ellipsis appears. Once the ellipsis appears, you tap the ellipsis to open the text box.

URLs in Applet Fields

Tapping on a field that contains a URL value launches the URL in PocketIE.

Navigating a List of Records in the Siebel Handheld Application

Within the view you select from the Show drop-down list, you can view detailed information on a record. When records are displayed in a list, you can drill down or drill across the record to get additional details.

Drilling down occurs when you tap a hyperlink in a record and are taken to another view within the current screen. For example, if you are in the Accounts screen, and tap the Account Name hyperlink, you navigate to the details for that account. You do not leave the Accounts screen; you just move deeper into it.

NOTE: Hyperlinks are indicated by blue underlined text as long as they are set up this way in User Preferences. For more information see "Setting User Preferences in the Siebel Handheld Application" on page 118.

Drilling across a record occurs when you tap a hyperlink in a record and are taken to another view in a different screen. For example, tapping a hyperlink in the Accounts column of a contact record in the Contacts screen takes you to the record for that account in the Accounts screen.

Dynamic drilldown occurs when you tap a hyperlink in a record and are taken to another screen or just another view. Where you navigate to depends on the content in the drilldown field. For example, if you tap the hyperlink in the Account field of the Activities Details list, you navigate to the Accounts Details view. If you tap the SR # field, you navigate to the Service Reqs Details view.

History Arrows

To navigate back to views you have recently displayed, tap the Back button. After you have used the Back button, you can use the Forward button to navigate back to your original screen and view.

NOTE: The History arrows only allow you to navigate between views. These arrows do not allow you to recover data that may have been entered or deleted in another view.

Record Navigation Buttons

Use the Record Navigation buttons to move back and forth though a set of records. The navigation buttons are described in Table 30.

Table 30. Record Navigation Buttons

Button	Description
	Navigates back to the previous record in the list or form.
	Navigates forward to the next record in the list or form.
	Navigates back a view based on the navigation history. You can define which views are part of the history. If a view is set to not be recorded as part of the history, the back button won't go back to that view.

Entering Data with the Siebel Handheld Application

There are several different methods for entering data in the Siebel Handheld application. Tap the up arrow next to the Input Panel button to select a method. See the user guide for your device for information on using the different input methods.

Working with Columns in the Handheld Client

Data is displayed in lists and forms in your Siebel Handheld application. You can organize and work within it in a number of ways to better suit your needs.

There are several ways in which to organize columns in a list. You can:

- Sort data by up to three columns
- Resize columns
- Change the order in which the columns appear
- Show and hide columns
- Lock columns for horizontal scrolling

You can sort records in a list by tapping in the column header of the column in which you want to sort the records. If the green column sort indicator points up, the column is sorted in ascending order. If it points down, the column is sorted in descending order.

If you need to sort by more than one column, you can open the Sort Order dialog box to sort up to three columns at one time. Open the Sort Order dialog box by choosing View > Sort from the application-level menu.

Resize columns by tapping and dragging the divider between the column headings until the column is the desired size.

You can change the order of columns using the Columns Displayed dialog box. Open the Columns Displayed dialog box by choosing View > Columns.

Show and hide columns using the Columns Displayed dialog box. In the Available columns list, select the items you want to show and tap the arrow to move the selected items to the Selected columns list. To hide columns, select the items and tap the arrow to move the selected items to the Available columns list.

Lock or unlock columns by tapping and holding the column header.

Finding and Querying Data in the Siebel Handheld Application

You can search for data using the Find or Query functionality in your Siebel Handheld application.

About Find in Handheld

Use Find to do a simple search for a record. You can access Find by choosing Edit > Find View > Find from the application-level menu. This launches the Find dialog box, in which you can enter your search criteria. The asterisk wildcard function (*) can be used in the search criteria. Figure 4 shows an example of the Find dialog window with the following fields: Look In, Last Name, Type, Last Call Date, Primary Specialty, and Rep Specialty.

NOTE: There may be a slight variation in your version of Siebel software, and the Find dialog box may not look exactly as shown here.

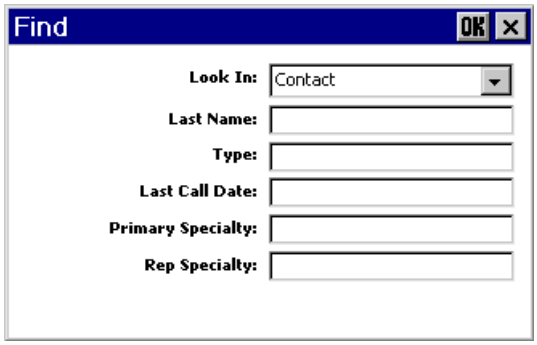

Figure 4. Find Dialog Box

About Asynchronous Query

Server Find, or Asynchronous Query is enabled, by default, in Siebel Service Handheld. Asynchronous Query utilizes Siebel Store-and-Forward Messaging to transmit data, and allows you to access data not available in the Siebel Handheld application's local database. If a user cannot find desired information on a Handheld application's local data store, the user can send a query in a format of a request message to the target server. The server receives the request message, performs the query, and returns the query results in a data message to the requesting Siebel Handheld application. Later on, the data message arrives at the Siebel Handheld application and its information is inserted into a temporary table on the Siebel Handheld application. The user can then review the query results.

About Queries in Handheld

Use Query to locate a number of records containing a specific set of criteria. There are a number of ways in which to perform query tasks in your Siebel Handheld application.

Creating a Query

To begin a query, you can choose View > Query > New... from the application-level menu, or you can tap the New Query button on the toolbar. Both provide you with a blank form or list where you can enter your query criteria, depending on where you are in the application when you invoke the command. The asterisk wild card function (*) can be used in the search criteria.

Executing a Query

After you have created the query, you can execute it by choosing View > Query > Run... from the application-level menu or by tapping the Execute Query button in the toolbar.

Refining a Query

You can refine your current query by choosing View > Query > Refine from the application-level menu.

Printing with the Siebel Handheld Application

You may print from any view in the Siebel Handheld application that has been configured to support printing. When you print from the handheld, a document that has been configured for the view is printed. You can print directly to a printer, or, if you are not connected to a printer at the time, you can queue your print jobs and wait until you have access to a printer to print your documents.

NOTE: When you print from a view, you are not necessarily printing what is visible on the screen at the time. Every field on the screen may not be relevant for the document. The document may contain only a portion of what is in the view, and it may also contain data that is not viewable at the time. A separate applet, which often cannot be viewed in the user interface, is configured specifically for printing.

The print templates used to generate the printed are installed during the time of setup.

For a list of supported printers, see *System Requirements and Supported Platforms* on Siebel SupportWeb.

Print from the Siebel Handheld application by choosing File > Print. If you are connected to a printer, the print job runs. If you are not connected to a printer, the print job is sent to a print queue. View the print queue by choosing File > Print Queue.

Exporting Data with the Siebel Handheld Application

You may export data from any view in your application in HTML format or as tab delimited text and save it to a file. This file can be viewed in applications such as Microsoft Pocket Word, Microsoft Pocket Excel, and Internet Explorer.

To export data, select File > Export to open the Export dialog box.

In the Export dialog box, you may specify the following:

- **Format.** Specify either HTML or Tab delimited text file.
- **Export.** Select Only active applet to export only the data in the active applet. Select All applets in view to export the data in all visible applets.

- **Output file name.** By default, the file is saved in the temp directory with the filename "output." You may tap Browse to specify a different location or filename.
- **Paste Output to Clipboard.** Select the checkbox to save the data to the clipboard.
- **Open file.** Select the checkbox to automatically open the file. When you tap OK in the Export dialog box, the data is exported to the specified file, and the file is automatically opened.

  NOTE: You must specify a valid output format for the application. See Table 31 for the correct formats to specify for the supported applications.

Table 31. Export Data Options

To export data to	Do the following
Microsoft Word	Set Format to "Tab delimited text file."
Microsoft Excel	Select the "And paste output to clipboard" check box. The Output format may be set either to HTML or "Tab delimited text file." Paste the data from the clipboard into an Excel spreadsheet.
Microsoft Internet Explorer	Set Format to HTML.

Setting User Preferences in the Siebel Handheld Application

You can customize aspects of your Siebel Handheld application from the User Preferences dialog box. Access the User Preferences dialog box by choosing View > Display... from the application-level menu.

In the User Preferences dialog box you can:

- Set a default startup screen.
- Set number of maximum history threads to be saved.

  CAUTION: Do not change the default setting for Maximum threads saved. This setting is optimized for the application.
- Set the maximum percentage of the display area that the parent applet can take up.
- Change font size.
- Customize list spacing, column width settings, and grid lines.

Backing Up and Restoring Data with the Siebel Handheld Client

It is recommended that you back up your data to an external device—for example, a CompactFlash card or a Secure ID card. If the handheld device is damaged, the backup may not be retrievable. If you store your backups in RAM, depending on the size of your application and data extract, you may decrease the amount of memory available for the application, and your application performance may suffer.

Automatic Backup After Synchronization

The automatic backup feature must be enabled in the application by your System Administrator. If the backup feature is enabled, a backup is automatically created at the end of every synchronization. This way, you have an up-to-date snapshot of the database.

CAUTION: You must never synchronize your data to the handheld device and then restore the database using an older copy of the database. If you do, the next time you synchronize, the application tries to rewrite changed transactions. This can cause data integrity problems. Therefore, when the database backup feature is enabled, a database backup occurs automatically at the end of each synchronization.

Keep in mind that enabling automatic backup makes the overall time to synchronize somewhat longer. The size of the database and the speed of the CompactFlash card also affect the time it takes to complete synchronization.

For more information, see "Enabling Handheld User Database Backup" on page 93.

Backing Up Data

The following procedure describes how to back up the database on the handheld device.

To back up the database

1. From the application-level menu, choose File > Backup.

   NOTE: If the Backup menu item is disabled, contact your system administrator to have backup enabled in your application.

   A dialog box appears, telling you that backing up the data will restart the application and prompts you to confirm that you want to continue.

2. Tap Yes to start the backup.

   A status dialog box appears indicating that the backup is in progress and may take a few minutes. The Handheld configuration files and data is copied to the backup location specified in the setup.ini file. Typically, the backup location is a CF or SD card.

   A dialog box appears telling you whether the backup succeeded and asks if you want to return to the application.

3 Tap Yes to return to the application.

NOTE: If the backup fails, consult your system administrator.

Restoring Data from a Backup

In order to restore the application to its previous state, the user should contact the system administrator. Any work performed on the handheld device since the last backup will be lost.

To restore the database

1. Remove the external flash card from the damaged device and place it in a new handheld device where the application has been installed.
2. Navigate to the \Program Files\Siebel Handheld folder.
3. Start the restore by tapping the BackupUtility file twice.

   A dialog box appears, asking you to confirm that you want to restore the device from a previous backup.
4. Tap Yes.

   A status dialog box appears, indicating that the restore is in progress. The backup utility restores the files to the state when the backup was performed. A dialog box appears telling you if the restore was successful. If successful, it asks you to confirm that you want to restart the Siebel Handheld application.
5. Tap Yes to start using the application.

Synchronizing Data on the Siebel Handheld Device

You must synchronize the data on your handheld device:

- After you install or upgrade the Siebel Handheld application.
- Before uninstalling the Siebel Handheld application.
- On a regular basis to keep your data synchronized with Siebel Server

The sections that follow describe how to synchronize using Direct Server Sync (DSS) and Direct Server Sync via Proxy (DSSvP), and provide some troubleshooting tips.

Using Siebel Handheld Synchronization with the Handheld Client

Siebel Handheld Synchronization differs depending on whether users synchronize through Companion Sync or Direct Server Sync. The following topics show you how to launch and use Handheld Synchronization for each method.

NOTE: A given user can only synchronize using one method. For example, it is not possible to install a Siebel Handheld application that a particular user will synchronize both to the Siebel Mobile Web client database through Companion Sync and also to the Siebel Server database directly through Direct Server Sync.

Companion Sync can also be performed in the method described for Direct Server Sync via Proxy in "Using Direct Server Synchronization with the Handheld Client" on page 121.

To perform Companion Sync from the handheld device

1. From the application-level menu, choose File->Synchronization.
2. In the Sync window select the "Enable Remote Sync" checkbox. It is disabled by default.

   For information on changing the parameter value for EnableRemote in the setup.ini file, see "Siebel Handheld setup.ini Parameters" on page 90.
3. See the steps for synchronizing using DSSvP in the procedure "To synchronize with the Siebel application server" on page 121.

Using Direct Server Synchronization with the Handheld Client

There are two methods for synchronizing between the handheld device and the Siebel application server:

- Direct Server Sync (DSS) provides synchronization between the Siebel Handheld application and the Siebel application server.
- Direct Server Sync via Proxy (DSSvP) provides synchronization between the Siebel Handheld application and the Siebel application server, using the desktop or companion PC to provide the network connection between the handheld device and the server.

To synchronize with the Siebel application server

1. If you are synchronizing using DSS, create a direct network connection through a modem or a direct LAN connection.

   If you are synchronizing using DSSvP, establish an ActiveSync connection between the handheld device and the PC.

2. From the application tap File > Synchronize.

   The Siebel Handheld Sync dialog box appears.

3. Enter your user name and password.
4. Tap Start.

   The Siebel Handheld Sync dialog box appears.

5. Tap OK to exit Siebel Handheld Sync, and the application will launch automatically.

   NOTE: If synchronization errors occur, an error dialog box appears.

Synchronization Troubleshooting with the Siebel Handheld Client

This section lists potential error messages that are associated with synchronization. In addition, this section helps to understand transaction errors.

General Synchronization Errors

There are a number of different errors that can occur when synchronization does not complete successfully, for example:

- Faulty wire connection
- Server crash
- Insufficient memory
- Login error

When these errors occur, the user should exit Siebel Handheld Sync and launch the application again.

NOTE: When you restart Siebel Handheld Sync, you may see an error message that says that another instance of syncmanager.exe is running. You are asked whether you want to terminate the first instance or not, and you should choose Yes.

Transaction Conflicts

Transaction conflicts are minimized when customers adhere to the standard configuration practices outlined in this document. However, even when proper configuration practices are followed, conflicts can occur for a variety of reasons. For example, if a field value is updated on both the handheld client and the server between synchronization events, a transaction update conflict occurs. In this instance, the user is presented with an error dialog after synchronization has completed.

A transaction consists of one or more smaller transactions called mini-transactions. For example, if you update several fields in a record, the update to the record counts as a single transaction, and each update to a field is a mini-transaction. Individual fields are skipped when update conflicts occur. Changes to any field that has been updated on the server since the last sync (perhaps by another Handheld user) are discarded (but logged and returned as errors to the user). All non-conflicting fields normally are updated. Except for field-level conflicts, which are skipped when they occur, a transaction is applied in its entirety, and at this point will either succeed or fail in its entirety.

Improving Siebel Handheld Application Performance

If you find that the performance of the Siebel Handheld application is very slow or begins to degrade, the following can help to improve the application performance:

- Verify that you are using a Siebel supported device for your application. See *System Requirements and Supported Platforms* on Siebel SupportWeb.
- When you run the Siebel Handheld application, it is recommended that you do not run any other applications. By doing so, you maximize the amount of memory that is available to run the application.
- Increase the amount of memory that is allocated for running applications.
- Installing the binaries to a CompactFlash card. For more information, see "Setting Up Siebel Handheld Application Installation on a CompactFlash Card" on page 94.

To increase application memory

1. From the Start menu, select Settings.
2. Select the System tab and tap Memory.
3. Move the slider bar so that more memory is allocated to Program Memory, then tap OK.

10 Using Service Handheld

This chapter includes the following topics:

Siebel Service Handheld User Scenario

This scenario provides an example of a workflow process performed by field technicians using Siebel Service Handheld. Your company may follow a different process flow according to its business requirements.

Each morning, the field technician synchronizes Siebel Service Handheld with the Siebel Server. After the synchronization is complete, she reviews her activities for the day, determines what parts she needs for her jobs, and verifies that she has the required parts. She records that she is en route to her first visit. When she arrives at the customer site, she records the time of her arrival and reviews the instructions and steps for the first activity. She checks off the steps as she completes them and records the starting and ending times for each step. She takes a reading and records the measurement. She needs to replace a part, and she records that the part is coming out of her trunk and generates an order for the movement. She completes all the steps for the activity, records her time, enters her expenses, and generates an invoice for the billable expenses, time, and parts. She closes out the activity and continues on to the next activity.

The procedures for each of these activities is covered in the following sections:

- "Preparing for the Day Using Siebel Service Handheld" on page 126
- "Preparing for a Visit with Siebel Service Handheld" on page 129
- "Performing the Job with Siebel Service Handheld" on page 130
- "Closing Out the Visit with Siebel Service Handheld" on page 133

About Siebel Store-and-Forward Messaging

Siebel Store-and-Forward Messaging and Siebel Service Handheld work together allowing you to perform critical business processes. A mobile technician using the Siebel Service Handheld application together with Siebel Store-and-Forward Messaging can receive a dispatched activity, acknowledge the activity, update the activity's status, move parts, or debrief the assigned activity. During this process, Siebel Store-and-Forward Messaging enables the mobile user to communicate all information, in real or near-real time, with the back-end Siebel application server.

For more information on Siebel Store-and-Forward Messaging, see *Siebel Store-and-Forward Messaging Guide*.

About Asynchronous Query in Siebel Service Handheld

Asynchronous Query allows you to use your Siebel Service Handheld application to access data not available in the application's local database. Asynchronous query uses Siebel Store-and-Forward Messaging to transmit data. If a mobile technician cannot find particular information on her Siebel Service Handheld application's local data store, she can send a query formatted as a request message to the target server. The server receives the request message, performs the query, and returns the query results in a data message to her Siebel Service Handheld application. When the data message arrives at the Siebel Handheld application, its information is inserted into a temporary table on the Siebel Handheld application. An alert informs the mobile technician that a message has arrived, and she can then use the Asynchronous Query view to review the results of her query. For information on setting up and using asynchronous query, see *Siebel Store-and-Forward Messaging Guide*.

Preparing for the Day Using Siebel Service Handheld

Before the field technician starts her day, she typically does the following:

- **Reviews an activity.** Before starting out for the day the field technician reviews the details of the activities assigned for that day. If you are using Siebel Store-and-Forward Messaging to receive assigned activities, see *Siebel Store-and-Forward Messaging Guide* for a guide to setup and usage.

  CAUTION: Siebel Service Handheld application filters allow you to review an activity that you create on the date it is created, when no Planned Start date is indicated. If you change the Planned Start date for the activity, you will see the following error message: "Error occurred while finding the prior place in the system". The activity will not be displayed in the Activity list view. The activity will instead be uploaded to the server database during the next handheld synchronization.

- **Adds a note for a contact.** The field technician adds a contact note for one of her contacts.

- **Orders parts for an activity.** Before going on a customer visit, the field technician verifies and orders any required parts and tools for the visit. If the part is not available in the local database of the Service Handheld application, you can send an asynchronous query to the back-end Siebel application server. For more information on asynchronous query setup and usage, see *Siebel Store-and-Forward Messaging Guide*.
- **Accepts and declines activities.** The field technician reviews the activities scheduled for the day and, after reviewing the activities, accepts or declines each activity.
- **Records that the field technician is en route to a customer visit.** As the field technician heads out to the customer site, she records that she is en route to the customer site.

The following procedures are performed from the Activities screen. From the Activities view in the Activities screen, select the activity. Once the activity is selected, the following tasks may be performed in any order. The tasks are as follows:

- "To review an activity" on page 127
- "To add a note for a contact" on page 127
- "To order parts for an activity" on page 128
- "To accept and decline activities" on page 128
- "To record that you are en route to a customer visit" on page 129

To review an activity

1. Navigate to the Activities screen.

   The Activities list appears, displaying the activities for the day by the planned start time, in ascending order.

2. In the Activities list, tap the hyperlink in the Activity # field.

   The Activities detail form appears.

3. In the Activities detail form, view the details for that activity by tapping the hyperlink in the following fields:

   - SR # (Service Request #). Tapping this hyperlink allows you to see the Service Request detail form.
   - Account. Tapping this hyperlink allows you to see the Account detail form.

   From the detail view for each field, tap the arrow button to return to the Activities Details form.

To add a note for a contact

1. In the Activities list, tap the hyperlink in the Activity # field.

   The Activities detail view appears.

2. In the Activities detail form, tap the hyperlink in the Contact field.
3. In the Accounts detail form, tap Notes to see any notes for the contact.
4. In the Contact Notes form, tap New Note to add a new note.

5 From the Type drop-down list, select the type of contact note.

6 Enter your text in the Note field.

7 Tap Back to return to the Activities Details form.

8 Tap Back to return to the Activities list.

To order parts for an activity

1 In the Activities list, select an activity and tap Accept.

2 Tap the hyperlink in the Activity field.

The Activities detail form appears.

3 In the Activities detail form, select Recommended Parts & Tools from the Show drop-down list.

4 In the Parts & Tools list, tap Check Trunk.

The application checks the inventory in your vehicle trunk and returns the number of available parts. If the part is not available, it looks for a substitute part and adds a line item for the substitute part.

NOTE: If you are unable to locate the part on your local database, you may send an asynchronous query to your back-end Siebel application server database. For information on setup and usage, see *Siebel Store-and-Forward Messaging Guide*.

5 Tap Order Part.

If an order does not already exist for the part, the application generates an order.

NOTE: You may generate an order for a line item from the Recommended Parts & Tools view and the Move Parts view. However, once the order is generated, you cannot undo the order from these views. You must go to the Service Request screen for that order, and delete the order from the Orders view.

6 Tap the hyperlink in the Order # field, then in the Order Details list.

7 Tap the hyperlink in the Order Line # field.

8 After you have reviewed the order line details in the Order Line Details list, tap OK.

9 In the Order Details list, tap Print Order, then tap OK.

To accept and decline activities

1 Navigate to the Activities screen.

2 In the Activities list, select the activity and do either of the following:

- Tap Accept to accept the activity.
- Tap Decline to decline the activity.

To record that you are en route to a customer visit

1. Navigate to the Activities screen.
2. Select the activity in the Activities list and tap En Route.

Preparing for a Visit with Siebel Service Handheld

Before a field technician starts an activity, he or she performs a number of tasks:

- Records the time of arrival at the customer site
- Reviews the instructions and steps for the activity
- Reviews the recommended parts and tools

From the Activities view in the Activities screen, select the activity. Once the activity is selected, the following tasks may be performed in any order.

To record the time of arrival

- In the Activities list, select the activity and tap Arrive.

  NOTE: You must have already accepted the activity (by tapping Accept) and recording movement (by tapping En Route) to successfully record arrival.

To review instructions and steps

1. Select an activity in the Activities list.
2. From the Show drop-down list, select Instructions.
3. Review the instructions in the Instructions list.
4. Select an instruction in the Instructions list.
5. From the Show drop-down list, select Procedures.
6. In the Procedures list, review the procedures.

To review recommended parts and tools

1. Select an activity in the Activities list.
2. From the Show drop-down list, select Recommended Parts & Tools.
3. Review the recommended parts and tools for the job.

Performing the Job with Siebel Service Handheld

While performing an activity, the field technician performs the following tasks:

- **Checks off completed steps.** As the field technician completes each step in the activity, she marks the steps as done.
- **Takes a reading.** For tasks such as taking a utility meter reading or measuring hydraulic pressure, the field technician records a reading.
- **Installs and replaces parts.** During a visit to a customer site, the field technician often needs to install and replace parts. There are two types of parts that can be replaced: parts with and without serial numbers.

From the Activities view of the Activities screen, she selects an activity. Once the activity is selected, the procedures can be performed in any order.

To check off completed steps

1. Select an activity in the Activities list, and from the Show drop-down list, select Time.
2. Start tracking your time by tapping the Start button and pick a type.
3. From the Show drop-down list, select Procedures.
4. Select the first step in the Procedures list.
5. When you have completed the step, select Done.
6. Enter any comments in the Comments field.
7. Complete the preceding steps for any remaining steps.
8. When you have completed the activity, select Time from the Show drop-down list.
9. Tap End to record the ending time for the activity.

To take a reading for a measurement

1. Select an activity in the Activities list.
2. Tap the Activity # hyperlink to drill down to the Activity detail form.
3. Tap on the Asset # hyperlink to drill down to the Asset detail form.
4. Tap on the Measurements button.
5. Tap on a Measurement hyperlink to drill down to a measurement.
6. Tap on Take Reading button to take a reading for the measurement.

Working with Non-Serialized Parts in Service Handheld

The following procedure describes how to install a non-serialized part.

To install a non-serialized part or product

1. Select an activity in the Activities list.
2. From the Show drop-down list, select Move Parts.
3. In the Move Parts list, tap Move.

   You can move a part manually by using the following steps, or use a barcode scanner to scan the part to be moved into the Move Part form.
4. In the Move Part form, tap the Product field.

   The Pick Product list appears.
5. In the Pick Product list, select a product and tap OK.
6. Tap the Asset # field.

   The Pick Asset list appears.
7. Choose an Asset number, then tap OK.
8. In the Activities detail view, check Billable, then tap OK to return to the Move Parts list.

Process for Working with Serialized Assets in Service Handheld

To install and replace a serialized part, perform the following three steps:

- "To remove a serialized asset" on page 131
- "To generate an RMA" on page 132
- "To install a serialized replacement part" on page 132

To remove a serialized asset

1. Select an activity in the Activities list.
2. From the Show drop-down list, select Move Parts.
3. In the Move Parts list, tap Move.
4. In the Move Part form, tap the fields and do the following:

Tap this field	Then do this
Product	Select a product from the picklist or scan the product using a barcode scanner.
Status	Select Defective from the picklist.
Asset	Select an asset from the picklist or scan the asset using a barcode scanner.
From	Select Customer from the picklist.

Tap this field	Then do this
To	Select Trunk from the picklist.
Comments	Add any comments.

5 Tap OK to return to the Move Parts list.

6 In the Move Parts list, tap Commit.

After removing a serialized part, you generate an Return Merchandise Authorization for the part. The procedure for generating an RMA assumes you have completed the procedure for removing a serialized part.

To generate an RMA

1 In the Move Parts list, tap RMA.

2 Tap the hyperlink in the Order # field.

3 In the Order Details list, tap the hyperlink in the Line field for the first record.

4 Review the order line item details in the Order Line Details form, then tap OK.

5 In the Order Details form, tap Print Order.

6 Tap Back to return to the Move Parts list.

After generating an RMA, you are ready to install a serialized replacement part. The procedure for installing a replacement part assumes you have generated an RMA for the defective part.

To install a serialized replacement part

1 From the Move Parts list, select Move Parts from the Show drop-down list.

2 Tap Move in the Move Parts list.

3 In the Move Part form, tap the fields and do the following:

Tap this field	Then do this
Asset	Select an asset from the picklist.
Status	Select Good from the picklist.
Comments	Add any comments.

4 Check Billable, then tap OK to return to the Move Parts list.

5 In the Move Parts list, tap Commit.

Closing Out the Visit with Siebel Service Handheld

At the end of the visit, the field technician completes the following tasks to close out the visit:

- Checks off any remaining steps
- Tracks billable time for the activity
- Records expenses
- Generates an invoice for the activity
- Closes out the job

Each of these tasks are performed from the Activities view of the Activities screen.

To check off the remaining steps

1. Select the activity in the Activities list.
2. From the Show drop-down list, select Procedures.
3. Select the remaining steps.

To track billable time

1. Select the activity in the Activities list.
2. From the Show drop-down list, select Time.
3. In the Time list box, select the record you created to start tracking your time.
4. Tap End.

   The ending time is entered and the time is calculated and entered in the Hrs field. The Billable field is selected by default.

You may also delete the time in the Start and End fields and manually enter the time in these fields.

To record expenses

1. Select the activity in the Activities list.
2. From the Show drop-down list, select Expenses.
3. In the Expenses list, tap New Expense.
4. Tap the Type field, then select an item from the Type drop-down list.
5. Tap the Amt field, specify an amount in the Enter Amount field, then tap OK.
6. Select Billable.
7. Tap the Comments field and enter any comments.

To generate an invoice

1. Select the activity in the Activities list.
2. From the Show drop-down list, select Invoices.
3. In the Invoices list, tap Auto Invoice.

   A record is inserted in the list.
4. Tap the Sign button to capture a signature
5. In the signature capture view, tap the New Signature button
6. From the Last Name pick contact list, pick a contact
7. Tap the ellipsis (...) in the Signature field.

   The signature field appears in the bottom of the view.
8. Capture a signature from the customer and tap on the Accept button to accept the signature
9. Tap Back to the Invoice list, then tap the Print Invoice button.

To close the job

1. Select the desired activity in the Activity field.
2. Tap Finish.
3. Tap the hyperlink in the Activity # field.
4. In the Activities Details form, tap the % Complete field to open the numeric calculator.
5. Enter 100 and tap OK.
6. Tap the Resolution field and select an item from the drop-down list.

   For example, Answered Question, Issued RMA and so on.
7. Tap Back to return to the Activities list.

Performing Miscellaneous Tasks with Siebel Service Handheld

The following ad hoc tasks are performed by the field technician as needed:

- Reviews customer history
- Logs a new service request
- Generates a new sales order
- Adds an activity
- Verifies inventory updates

Reviewing Customer History with Service Handheld

A field technician may want to review the history of service requests for a customer so that she is aware of the service history for the customer.

To review customer history

1. Navigate to the Accounts view of the Accounts screen.
2. Find the customer account you want to review.
3. From the Show drop-down list, select Service Requests.

   The Service Requests view appears showing all the service requests for the account.

Logging a New Service Request with Service Handheld

While out on a customer visit, the field technician discovers another problem requiring attention and logs a new service request for the customer. She performs the following tasks:

- Confirms the customer's asset
- Confirms the warranty coverage of the asset
- Confirms the service level entitlement
- Logs a new service request

To confirm asset and warranty coverage

1. Navigate to the Accounts view of the Accounts screen.
2. Select the account and from the Show drop-down list, select Assets.
3. In the Assets list, tap the hyperlink in the Asset # field.
4. In the Asset Details list, verify the applicable warranties.
5. Tap Back to return to the Assets list.

To confirm service level entitlement

1. Navigate to the Accounts screen.
2. From the Show drop-down list, select Entitlements.
3. Verify the entitlements in the Entitlements list.

To log a new service request

1. Navigate to the Accounts screen.
2. From the Show drop-down list, select Service Requests.

3 In the Service Requests list, tap Log New SR.

4 Complete the fields in the New SR form.

5 Tap OK.

Generating a Sales Order with Service Handheld

This procedure describes how to generate and print a sales order for a service request.

To generate a sales order

1 Navigate to the Service Reqs screen and select or query for the desired service request.

2 Select Orders from the Show drop-down list.

3 In the Orders list, tap Create New Order.

4 In the Service Reqs form, select Sales Order from the Type drop-down list.

5 In the Req Ship (Requested Ship Date) field, select a date from the pop-up calendar, and tap OK.

6 In the Order Details list, tap the Add Order Line button.

7 Tap in the Product, and select a product from the Pick Product pop-up list.

8 Complete the remaining fields in the Order Line Details view, the tap OK.

9 In the Order Details list, tap Print Order to print the order, then in the Service Reqs form, tap OK.

Adding an Activity in Service Handheld

A field technician can add an activity to an existing service request or she can add a sundry activity, such as making a bank deposit or taking her truck in for servicing, which is not associated with a particular account.

CAUTION: Siebel Service Handheld application filters allow you to review an activity that you create on the date it is created, when no Planned Start date is indicated. If you change the Planned Start date for the activity, you will see the following error message: "Error occurred while finding the prior place in the system". The activity will not be displayed in the Activity list view. The activity will instead be uploaded to the server database during the next handheld synchronization.

To add an activity to a service request

1 Navigate to the Service Reqs screen and select the desired service request.

2 From the Show drop-down list, select Activities.

The Service Reqs Activities view appears.

3 In the Activities list, tap Add New Activity.

The New Activity form appears.

4. Tap the Activity field, and select an activity type from the drop-down list.
5. Select the Status field, and select Not Started from the drop-down list.
6. Complete all fields in the New Activity form, and tap OK.

To add a miscellaneous activity

1. Navigate to the Activities view, and in the Activities list, tap Add New Activity.
2. In the New Activity form, select an activity type from the Activity drop-down list.

   For example, Administration, Call-Inbound and so on.
3. Select SR # and Account from the picklist.
4. Tap the drop-down arrow in the Status field and select Not Started from the picklist.
5. Enter any text in the Comments field.
6. Tap OK.

Verifying Inventory Update in Service Handheld

A field technician periodically verifies that the inventory in the van matches the inventory that is recorded in the system.

To verify inventory updates

1. Navigate to the Browser screen.
2. Query for individual product you want to verify.
3. Tap the hyperlink in the Product field of the product you are verifying.
4. Compare the balances in the Availability field with your printed inventory report.

A Troubleshooting

This appendix addresses some of the common issues faced while deploying Siebel Handheld. The appendix presents troubleshooting tips in three areas:

- Installation
- Performance
- Log Files

The audience for this appendix is Siebel database administrators, system administrators, and application administrators.

Installation

Siebel Handheld supports a specific and limited set of devices, handheld OS, desktop OS, server OS, server databases, and languages. This range of devices and platform support varies widely across the range of Siebel Handheld releases and must be carefully checked to confirm support. See *System Requirements and Supported Platforms* on Siebel SupportWeb for your Siebel application for information on supported Handheld OS, desktop OS, server OS, and server databases.

Client Installation, Backup, and Restore

If you encounter problems while installing the Siebel Handheld application on client devices from a desktop PC, refer to the following checklist. Most installation problems are covered by this checklist.

- Confirm that the device is certified and supported. Refer to *System Requirements and Supported Platforms* on Siebel SupportWeb.
- Confirm that the device has 64 MB or higher of RAM with the majority of it available for the Siebel Handheld application.
- Make sure an ActiveSync connection is running (green icon). If not, check your computer to make sure that the serial or USB port is activated and functioning correctly.

- If previous versions of Siebel Handheld were installed on the device, make sure you uninstall the application. See "Uninstalling the Siebel Handheld Application" on page 107.

Siebel Handheld binaries may also be installed to a CompactFlash card to conserve (RAM) memory. The application will still be loaded into memory when it executes. To install on the CompactFlash card, set the ApplicationBinariesLocation and ApplicationDataLocation parameters in the setup.ini file. See "Setting Up Siebel Handheld Application Installation on a CompactFlash Card" on page 94.

Siebel Handheld (application only - no data) may also be copied onto a CompactFlash card to aid in deployments. The initial backup image should be made of a device that has *not* done an initial synchronization. An initial synchronization will still be required by each user to receive the configuration and data. For information on backup utilities, see the documentation accompanying your handheld device.

It is also possible to back up Siebel Handheld on a PC using the Backup/Restore feature within ActiveSync.

Synchronization

Synchronization problems can be caused by incorrect settings in the Siebel Handheld setup.ini or .cfg files. Be sure that application names, data sources, and .srfs are correct.

Make sure to set the SyncURL setting in the client setup.ini file for all synchronization methods. Also, be sure to grant the user visibility to these views in the Administration - Application > Responsibilities view.

NOTE: Check the log files and confirm the accuracy of the generated connect string.

Direct Server Sync Checklist

Use the following checklist to troubleshoot problems you encounter using Direct Server Sync.

- Did you install all client and server components successfully?
- Is there a PDA object manager (OM) running on the Siebel Server?
- Did you bounce the IIS, SWE, and Siebel Server after installation?
- Do you have network access to the URL for DSS? Can you ping the IIS server from inside and, if applicable, from outside the firewall?
- Can you successfully run a SWE command?
- Do you have the correct SyncURL defined in the setup.ini file?
- Is the network connection working from the device? Try Pocket Internet Explorer.
- If you have problems logging in, check to make sure that you can use the same credentials to log into the desktop client.
- Remember to keep a 1:1 ratio of users to devices. Only a single user may log in to sync each device. Failure to do so causes data corruption.

To troubleshoot Direct Server Sync configuration errors

1. Turn on Handheld Sync logging to 5 and look for more error details in the Siebel Handheld Synchronization OM log. For more information, see "Configuring Direct Server Sync Logging Levels" on page 86.
2. Use the Validate function in Siebel Tools to look for a configuration error.
3. Remove handheld views from user responsibility until synchronization passes the error point.
4. Remove DefaultBusObjs from Administration - Mobile > Application Administration > Settings until synchronization passes the error point.

Re-synchronizing Problems

Generally if, for whatever reason, synchronization is terminated, the end user is prompted with a message and the synchronization process ends. In this situation the synchronization session is allowed to close. However, if the connection with the server is unintentionally broken, the session may not have a chance to close before the connection is broken. Therefore, if the end user tries to immediately synchronize again, he may get an error message and may not be able to synchronize. Should this occur, the end user should wait for a period of time (generally, 15 minutes) to allow the session to time out and terminate. Once the session terminates, the end user should be able to synchronize again.

Log Files

This section includes the following topics:

Direct Server Sync Log and Output Files

The name of the Direct Server Sync Log is the OM name plus the task ID for that log session. The log is found in the log subdirectory under the Siebel root directory. Depending on the amount of detail enabled, this file can give a good perspective of what has happened during synchronization regarding the Siebel OM.

To assist Siebel Technical Support, collect the following files when logging a Siebel Handheld service request. The following files are found on the machine where Siebel Server is installed:

- All server sync user files in the *Siebel Root*\siebsrvr\hhsync*application directory**user id**node id* directory.
- Server application event log extract.
- Server system event log extract.

- All files in the *Siebel Root*\siebsrvr\log directory including CompName_task.log.

  where CompName is the name of an application server component, and task is a Siebel task number. Each sync session is assigned a new Siebel task, so there will be many of these. An example "CompName" value is "SalesCE_enu" (note that language is part of the component name itself).
- All files from all subdirectories under the *Siebel Root*\siebsrvr\logarchive directory.
- Server perfmon log files (showing memory, network, disk performance), if available.

The following files are found on the machine where the Web Server (that is, IIS) is installed:

- SWE log files from *Siebel Root*\eappweb\log (all files).
- IIS server application event log extract.
- IIS server system event log extract.
- IIS server perfmon log files (showing memory, network, disk performance), if available.

Handheld Device Log

The siebel_ log.txt file is found in the /Program Files/Siebel Handheld directory on the handheld device. It is controlled by reg key DefaultLogLevel, with values 0-4. For more information on values, see "Siebel Handheld setup.ini Parameters" on page 90.

The default value of the reg key during installation is 2. This file logs all of the following:

- Database objects that were imported during synchronization.
- All SQL statements generated by the application. You must set a parameter in the setup.ini file to turn this log on. DefaultLogLevel=3 and 4 will log SQL statements to the device log. See "Siebel Handheld setup.ini Parameters" on page 90 for more information about setting the parameter.
- All system error messages. You must set a parameter in the setup.ini file to turn this log on. DefaultLogLevel=4 will give the most detailed log with debug info. See "Siebel Handheld setup.ini Parameters" on page 90 for more information about setting the parameter.
- All of the messages displayed on the Sync Manager dialog box as well as some basic information about synchronization, such as data source and connect string. You can set the logging level for this file using the DefaultLogLevel parameter in the setup.ini file.

Logging Handheld Synchronization Errors

Siebel Handheld Sync logs all the strings that the user sees in the sync message window as well as many informational messages that do not get displayed in the UI. These informational messages are used to help track down where an error occurred. Siebel Handheld Sync determines which messages get logged based on a value stored in the registry under the key DefaultLogLevel. The values range from 0 to 4; 0 logs all error messages and strings that the user sees, and 4 logs all data that Siebel Handheld Sync sends and receives.

If Siebel Handheld Sync fails, check the siebel_log.txt file immediately after the failure. If you need more detail, increase the logging level of DefaultLogLevel and run Siebel Handheld Sync again. For more information on changing DefaultLogLevel, see "Siebel Handheld setup.ini Parameters" on page 90.

You can determine when and where an error occurs during handheld synchronization by referring to siebel_log.txt and the Handheld Sync Object Manager log.

The device log, siebel_log.txt, shows the last HTTP command that was sent to server, for example, ExecuteLogin, Init, ProcessTransactionEx, ExtractDB, and so on. The Handheld Sync Object Manager log receives the request and invokes the corresponding method.

For example, in the device log (siebel_log.txt), the following indicates that the device is sending an "Init" request to server:

```
INFO0x6F46107E03/30/2005 16:58:27http ://PPP_PEER:2323/servicece/
start.swe?SWEExtSource=siebelpda&SWEExtCmd=Init&PatchLevel=7.7.2.2PDA%5b18356_1]
8.471262E-309NU&SyncFormat=CE0&SyncNodeId=&TimeStamp=1112230707

INFO0x6F46107E03/30/2005 16:59:20<Init ReturnCode="OK" Complete="100">
```

or, for Service SIA

```
INFO0x6F46107E03/30/2005 16:58:27http ://PPP_PEER:2323/servicece/
start.swe?SWEExtSource=siebelpda&SWEExtCmd=Init&PatchLevel=7.7.2.2PDA%5b18356_1]
8.471262E-309NU&SyncFormat=CE0&SyncNodeId=&TimeStamp=1112230707

INFO0x6F46107E03/30/2005 16:59:20<Init ReturnCode="OK" Complete="100">
```

The server log (Handheld Sync OM log, that is, CompNameObjMgr_enu_xxxxx.log), shows that the "Init" request is received by the server and the "Init" method is invoked:

```
ObjMgrBusServiceLogInvokeMethod402005-03-23 10:45:27Begin: Business Service
'Handheld Synchronization' invoke method: 'Init' at 10c2bc68

HandheldSyncLogDebug502005-03-23 10:45:27Request Init received
```

(Where "CompName" is the server component name of the handheld synchronization for the specific handheld application.)

In addition to the device log, customers should also check SalesCEObjMgr_enu_xxxxx.log in Siebel Server (for DSS). For more information on changing logging levels, see "Configuring Direct Server Sync Logging Levels" on page 86.

SQLTrace

Use SQLTrace for diagnostic purposes if you encounter serious configuration errors or performance problems and cannot otherwise diagnose the problem. The problem may quickly become apparent in the log entries from SQL tracing.

To enable SQL tracing

1. Log into srvrmgr prompt.

2. At the DOS prompt, run server manager (srvmgr) from the <siebsrvr/bin> directory: srvrmgr /g gateway /e enterprise /u sadmin /p sadmin
3. Run: `change evtloglvl ObjMgrSqlLog=4 for comp [OMName]`, where [OMName] = the OM name (for example, ServiceCEObjMgr_enu).
4. Log entries are written as described in "Log Files" on page 141.

   CAUTION: When SQL tracing is enabled, synchronization is severely and adversely affected. Therefore, it is recommended that you turn SQL tracing on for a limited time to diagnose a specific problem, then turn SQL tracing off.

End User Error Messages

In addition to errors found in the siebel_log.txt file, you might receive a Transaction Errors Reported message during synchronization. These errors are logged in TXNError.hh. The message appears on the handheld device in the Syncmanager Status box. You can double-click the message at any time to see the errors.

You can make the following settings in setup.ini:

TXNErrorLevel = 0 (no log information)

TXNErrorLevel = 1 (report txn errors in the client)

For more information on end-user messages, see "Synchronization Troubleshooting with the Siebel Handheld Client" on page 122.

Web Server Timeout Errors

If synchronization times are long due to the size of the database file or the speed of the network connection, your end users may encounter Web server timeout errors and synchronization failures. First try to resolve this problem by reducing the size of your extract and configuring your server performance. You can reduce the size of the extract by fine-tuning filters, or if appropriate filters have already been defined, select ones that will return less data. If after doing this, your end users continue to experience timeout errors, then increase the timeout parameter on IIS for the default Web Server.

Follow the procedure for your platform to change the timeout parameter.

To change the timeout parameter on Windows

1. From the Start menu, select Programs > Administrative Tools > Internet Services Manager.
2. From the Internet Information Services window, select the local machine, right-click on the Default Web Site, and select Properties from the pop-up list.
3. From the Web Site Properties window, select the Web Site tab.
4. Under the Connections heading, locate the Connection Timeout property and increase the number of seconds.

5 Click OK.

Handheld Device Timeout Errors

The following three parameters are stored in device registry. They control the device timeout. For more information on device timeout errors, see the following parameters in "Siebel Handheld setup.ini Parameters" on page 90:

- InternetOptionReceiveTimeout = 600000
- InternetOptionSendTimeout = 600000
- InternetOptionDataExtractionTimeout = 600000

Performance

This section discusses ways you can optimize your client performance, server performance, and network performance.

Client Performance

Client performance is affected primarily by the device capability (minimum 206 MHz processor and 64 MB RAM), configuration, and the quantity of data loaded onto the device. In general, screen-to-screen and view-to-view navigation times of 3 to 4 seconds or less should be considered normal.

It is strongly recommended that customers follow these general guidelines to optimize the performance of their handheld applications:

- Keep configuration under 30 views.
- Keep RML file size under 1.75 MB.
- Keep database files (dbfile.txt) under 2 MB.

  Use filters as outlined in the Siebel 2000 Handheld Synchronization Technical Note.
- Keep the number of records for each business component under 2,500.
- Avoid installing other applications on the device at the same time.
- Users should synchronize their data frequently.

Additional configuration information can be found in the Siebel Technical Note 405.

The Siebel Handheld client will benefit from periodic closing and restarting the application to release memory consumed by the handheld operating system. Use File > Exit in the menu to exit from the Siebel Handheld application.

If an external VGA driver is or has been loaded on the device in the past and is no longer needed, be sure to unload it, because it is known to have significant performance impacts on handheld devices.

Server Performance

Server performance affects the performance and scalability of Direct Server Sync processes. Customers should engage Expert Services to do a Scalability Review. The following guidelines are general in nature and pertain to the Siebel Handheld Sync components running on a Siebel Server:

- Use Performance Monitor to track memory and processor performance on the Siebel server.
- Make sure that the Servers (IIS, GW, OM and DB) are on the same high speed LAN segment.
- Siebel Handheld Sync components will stress Siebel Server and database performance because of the intensive interactions during large concurrent synchronizations.
- Estimate that 64 MB is used per Siebmtsh.exe instance.
- To calculate the theoretical maximum concurrent users:
  - Numusers=Server_Memory*threads_per_process_ratio/64 MB.
- Optimize the Threads per Process Ratio (Max Tasks/Max MT Servers) per Technical Note 405.
- Set Max Tasks=Max. number of concurrent users (for example, 150).
- Set MinMTServers=MaxMTServers (for example, 30 for 5:1 ratio, if this is determined to be appropriate).

To optimize and tune a Siebel DB server:

- Use Performance Monitor to track memory and processor performance on the DB server.
- Turn on SQL tracing and search for long-running SQL calls.
- Estimate 64 MB of RAM for each concurrent user.
- Consider creating indexes to tune the queries (only performed with the assistance of Siebel Expert Services).
- Consider using a RAID 0+1 disk array with an expanded number of spindles to optimize database performance.

NOTE: Make sure that DB Server settings are set to recommended values. Timeouts should be set at the Siebel Server. Be careful that the timeout is set longer than the longest SQL query.

Network Performance

The condition of the network affects Direct Server Sync performance and scalability. In general, a low-latency, high-bandwidth network is preferred with dedicated rather than shared network connections. The following are general network considerations:

- Make sure there is adequate bandwidth throughout the entire round-trip of network packets and estimate where bottlenecks will occur due to high concurrent load. For example, if 50 users are concurrently synchronizing 1 MB each, 50 MB of data must pass through the network. If this data is passing through a single leased 56 KB per second line, then this data will minimally take 500,000 KB/56 KB/sec=8928 seconds=148 minutes. In reality, a 56 KB per second line will offer only 30-40 KB per second average throughput, which causes further bandwidth constraints. Check the total available bandwidth of network providers (private or public ISP).
- Verify that you have low latency lines with ping round trips of less than 1 second (1000 ms) with no dropped packets.

- Verify if roundtrips occur with less than 10 hops.
- Check the actual throughput of network connections by performing simple file transfers with no Siebel software.
- VPN software from third parties accounts for 30 to 60 percent of additional packet overhead because of the encryption.

Network problems can usually be resolved by working with network providers to size network connections and optimize network routing. Consider the use of burst networks, quality of service contracts, private networks, and dedicated modem banks directly into a high-speed LAN environment.

B Screens and Views

Siebel Handheld Client supports a subset of the screens and views supported by the Siebel Web Client application. Other screens and views can be configured for your Siebel Handheld Client application if they are based on currently supported classes. Application developers and application administrators will find the information in this appendix useful.

If you configure screens and views based on unsupported classes, you may see anomalous results. For information on how to determine the classes used by other screens and views, see *Object Types Reference*.

Screens and Views

The screens and views provided with the Siebel Service Handheld application are shown in Table 32.

* These are hidden views and are accessible only by drilling down from another view.

Table 32. Siebel Service Handheld Screens, Views, and Business Components

Screen	View Name on UI	View Name in Siebel Tools	Business Component
Accounts	Accounts	SHCE Service Accounts View	■ Account
	Asset Details *	SHCE Service Account Asset Details View	■ Asset Mgmt - Asset ■ FS Asset Warranty
	Assets	SHCE Service Activity Assets List View	■ Account ■ Asset Mgmt - Asset
	Contact Details *	SHCE Service Account Contact Details View	■ Contact
	Contact Notes *	SHCE Service Account Contact Notes View	■ Contact Note
	Contacts	SHCE Service Account Contacts View	■ Account ■ Contact
	Details *	SHCE Service Account Details View	■ Account
	Entitlements	SHCE Service Account Entitlements View	■ Account ■ Entitlement Account

Table 32. Siebel Service Handheld Screens, Views, and Business Components

Screen	View Name on UI	View Name in Siebel Tools	Business Component
	New SR *	SHCE Service Account SR Details View	■ Service Request
	Readings *	SHCE Service FS Measurement Readings View	■ FS Asset Reading
	Service Requests	SHCE Service Account SRs View	■ Account ■ Service Request
Activities	Activities	SHCE Service Activity List View	■ Action
	Assets	SHCE Service Activity Assets List View	■ Action ■ Asset Mgmt - Asset
	Attachments	SHCE Service Activity Attachment View	■ Action Attachment ■ Asset Mgmt - Asset
	Details *	SHCE Service Activity Detail View (More Info)	■ Action
	Expense Reports	SHCE Service Activity Expense Report View	■ Expense Item ■ Expense
	Expenses	SHCE Service FS Expense View	■ Action ■ FS Expense Item
	Instructions	SHCE Service Instructions View	■ FS Instructions
	Invoices	SHCE Service FS Activity Invoice View	■ Action ■ FS Invoice
	Move Part *	SHCE Service FS Activity Part Movements (More Info)	■ FS Activity Parts Movement
	Move Parts	SHCE Service FS Activity Part Movements	■ Action ■ FS Activity Parts Movement
	New Activity *	SHCE Service Activity Entry View	■ Action
	Recommended Parts & Tools	SHCE Service FS Activity Recommended Parts Tools View	■ Action ■ FS Activity Recommended Parts_Tools

Table 32. Siebel Service Handheld Screens, Views, and Business Components

Screen	View Name on UI	View Name in Siebel Tools	Business Component
	Procedures	SHCE Service FS Steps View	■ Action ■ FS Activity Step
	Time	SHCE FS Time View	■ Action ■ Time Sheet Daily Hours
	Time Sheets	SHCE Service Activity Time Sheet View	■ Time Sheet Items ■ Time Sheet
Browser	Availability *	SHCE Service Part Browser Availability	■ FS Bucket Header - Part browser ■ FS Bucket - Part Browser
	Products	SHCE Service My Part Browser View	■ FS Bucket Header - Part browser
	Substitutes *	SHCE Part Browser Substitute View	■ FS Bucket Header - Part browser ■ FS Substitute Part Bucket - Part Browser
Service Reqs	Activities	SHCE Service Service Request Detail View	■ Service Request ■ Action
	Details *	SHCE Service Service Request View	■ Service Request
	Invoice Lines *	SHCE Service FS SR Invoice List View	■ FS Invoice Line Items
	Invoices	SHCE Service FS SR Invoice View	■ FS Invoice
	New Activity *	SHCE Service Service Request Activity Entry View	■ Action
	New SR *	SHCE Service Service Request SR Entry View	■ Service Request
	Order Details *	SHCE Service Order Line Items View	■ Order Entry - Order ■ Order Entry - Line Items
	Order Line Details *	SHCE Service Service Request Order Line Items View	■ Order Entry - Line Items

Table 32. Siebel Service Handheld Screens, Views, and Business Components

Screen	View Name on UI	View Name in Siebel Tools	Business Component
	Orders	SHCE Service Service Request Orders View	■ Service Request ■ Order Entry - Orders
	Service Requests	SHCE Service SR List View	■ Service Request

C Business Components and Classes

Siebel Handheld Client supports a subset of the business components and classes supported by the Siebel Web Client application. This appendix lists the supported business components, business component classes, and applet classes.

Supported Business Components for SiebelService Handheld

Table 33 lists the supported business components for Siebel Service Handheld.

Table 33. Supported Siebel Service Handheld Business Components

Supported Business Components
Account
Action
Agreement Entitlement
Asset Mgmt - Asset
Asset Mgmt - Asset - Sub Components
Contact
Contact Note
Currency
Employee
Entitlement Account
FS Activity Parts Movement
FS Activity Parts Movement Admin
FS Activity Recommended Parts_Tools
FS Activity Step
FS Activity Entitlement Service Detail
FS Asset Measurement Characteristics
FS Asset Reading

Table 33. Supported Siebel Service Handheld Business Components

Supported Business Components
FS Asset Warranty
FS Bucket
FS Bucket - Part Browser
FS Bucket Header - Part browser
FS Expense Item
FS Instruction
FS Inventory Location
FS Inventory Transaction
FS InvLoc Product
FS Invoice
FS Invoice Line Items
FS InvTxn Ledger Entry
FS PickList Inventory Txn Type
FS Product Inventory Category
FS Rate List Item
FS Service Detail Expense Bill Item
FS Service Detail Product Bill Item
FS Service Detail Time Bill Item
FS Substitute Part Bucket - Part Browser
FS Transaction Assets
FS Transaction Assets Alt
Internal Product
Order Entry - Line Items
Order Entry - Order Sales Items
Order Entry - Order Types
Order Entry - Orders
Price List
Price List Item
Project
PS Rate List
Service Request

Table 33. Supported Siebel Service Handheld Business Components

Supported Business Components
Time Sheet Daily Hours
Time Zone

Business Component Classes

NOTE: The business component classes for the handheld application are based on the classes for the Web Client. However, in most instances, because of the limitations of the handheld, only the subset of the class functionality that is required for the handheld is supported.

Table 34 lists the supported business component classes for Siebel Service Handheld.

Table 34. Supported Siebel Service Handheld Business Component Classes

Supported Business Component Classes
CSSBCActivity
CSSBCActivityPart
CSSBCActivitypartMvmt
CSSBCAssetReading
CSSBCAssetWrnty
CSSBCBase
CSSBCBusComp
CSSBCInvoice
CSSBCLineItem
CSSBCOrder
CSSBCPMAsset
CSSBCServiceRequest
CSSBCTmshtitem
CSSBCUser

Applet Classes

You may create additional screens and views for your handheld application. Any new screens and views must be based on classes and business components that are supported for the Service Handheld application. See Table 35 for the supported applet classes.

Table 35. Siebel Service Handheld Applet Classes

Supported Applet Classes
CSSFrameBase
CSSFrameList
CSSFrameListBase
CSSFramePopupCurrency
CSSFrameServiceRequest
CSSFrameSRActivity
CSSSWEFrameListPick

D User Properties and Methods

This appendix documents the handheld-specific user properties and methods for the Siebel Handheld application.

User Properties

Table 36. User Properties

Name	Comments
HandheldSyncPickMode	User property on a field of a business component. When set to NoFail, turns Extended Pick processing ON. See "Enabling Extended Pick Processing in Siebel Handheld" on page 100 for more information.
HandheldSyncInsertMode	User property on a business component. When set to FailOnError, enables insert failure processing. See "Insert Failure Processing in Siebel Handheld" on page 102 for more information.
OrderAccessControl	User property on the Order business component. When set to TRUE, all fields except Status are set to read-only for all users except the user who created the order.
OverridePopupVisibility	Overrides Popup Visibility Type on a business component. Use OverridePopupVisibility to change the visibility on a picklist without changing the Popup Visibility Type. See "Overriding the Popup Visibility" on page 74 for more information.
OverrideDefaultApplet	Overrides the default behavior for giving focus to an applet. The OverrideDefaultApplet user property is typically used with GotoViewNewRecord method. See "Applet Focus Behavior" on page 24 for more information.
View	Used with the GotoView method. Specifies the view to be displayed when navigating from one view to another. See "Applet Focus Behavior" on page 24 for more information.

Methods

Table 37 lists Business Components and details methods you can use for your Siebel Handheld application.

Table 37. Siebel Service Handheld Supported Methods

Business Component	Method	Description
Action	AcceptStatus	Sets the Status field of an activity to Acknowledged. Method is enabled when Status is not set to Acknowledged, On Hold, In Progress, Done, or Cancelled.
	ArrivedStatus	Sets the Start field of an activity to the current time. Method is enabled when Status = In Progress and the Start field is not set.
	EnRouteStatus	Sets the Status field of an activity to In Progress. Method is enabled when Status = In Progress.
	DeclineStatus	Sets the Status field of an activity to Declined. Method is enabled when the Status field is not set to In Progress, On Hold, Done, Cancelled, or Declined.
	SuspendStatus	Sets the Status field of an activity to On Hold. Method is enabled when Status = In Progress and the Start field has been set.
	ResumeStatus	Sets the Status field of an activity to In Progress. Method is enabled when Status = On Hold.
	FinishedStatus	Sets the Status field of an activity to Done, sets the End field to the current time, and sets the % Complete field to 100. Method is enabled when Status = In Progress and Start field is not empty.
	Start	Sets the Start field of an activity to the current time. Method is enabled when Status = In Progress and the Start field is not set.
	End	Sets the Status field of an activity to Done, sets the End field to the current time, and sets the % Complete field to 100. (Did not update % Complete field.) Method is enabled when Status = In Progress and Start field is not empty.

Table 37. Siebel Service Handheld Supported Methods

Business Component	Method	Description
FS Recommended Parts & Tools	OrderParts	Method is used in the SHCE Service FS Activity Recommended Parts Tools view, SHCE Service FS Activity Recommended Parts & Tools List Applet applet. For any record in the view, if there are insufficient parts—that is, Available Quantity is less than Recommended Quantity—executing this method orders additional parts. The method generates an AutoOrder for the selected products with the Recommended Quantity equal to the quantity in the order line item.
	CheckTrunk	Method is used in the SHCE Service FS Activity Recommended Parts Tools view, SHCE Service FS Activity Recommended Parts_Tools List Applet. Checks each record in the view and verifies the quantity (Qty field) against the inventory. It updates the Available Quantity (Avail) field with the number of items in the end user's trunk.
FS Activity Parts Movement	CommitPartMvmtClient	Method is used in the SHCE Service FS Activity Part Movements view, SHCE Service FS Activity Part Movements List Applet applet. This method commits a part movement—that is, it sets the Commit flag on the record, and once this flag is set, the Object Manager updates the inventory levels and the transaction table.
	OrderPartsRMA	Generates an order for a part movement and sets the order type to RMA.
	OrderPartsRMARepair	Generates an order for a part movement and sets the order type to RMA Repair.
FS Invoice	GenerateInvoice	Generates an invoice for an activity and creates line items for parts movement, time, expenses, entitlements, discounts, and totals. When the method is used in the Activity screen, the Action GenerateInvoice method is executed for a single activity. When the method is used in the Service Request screen, the Service Request GenerateInvoice method executes for each activity in the service request, it compiles one total for each line item, and generates one invoice for the entire service request.
Time Sheet Daily Hours	SetEndTime	Sets the end time in the SHCE FS Time View view, SHCE Activity Time Daily Hours List Applet applet.
	SetStartTime	Creates a new record and sets the start time in the SHCE FS Time View view, SHCE Activity Time Daily Hours List Applet applet.

E Print Tagging Language

This chapter describes the print tagging language that is used to create print templates for your handheld device.

The audience for this appendix is Siebel application developers.

"Overview of Siebel Handheld Print Tagging Language" on page 161

"Using Variables in Print Templates" on page 177

Overview of Siebel Handheld Print Tagging Language

You can create the print template file in any text editor.

- Tags are enclosed in angle brackets and identified with the keyword TAG, followed by a colon.

  For example: <TAG: ....>

- Parameters are specified with a keyword, followed by an equals sign. Arguments are in uppercase, surrounded by quotation marks.

  For example, Font = "FONT FACE"

- Values are in uppercase and lowercase, surrounded by quotation marks.

  For example: Font = "Helvetica"

- Tags may appear in any order in the file. The only required tag is the Title tag.
- Variables within a print template must be uniquely named. Variable names must be one-word strings and spaces are not allowed.

The following tags are used to specify the format of the print output, which applets are included, and the query that is executed:

- Applet—Specifies which applets are used in the document.
- Comment—Adds comments that do not appear in the final document.
- Divider—Adds visual lines to visually separate different parts of the document.
- Footer—Adds information such as page numbers, date, and time.
- Format—Specifies characteristics such as font, boldface, italics, and underline.
- Header—Adds information such as page numbers, date, and time.
- Page Break—Specifies a break in the text, forcing the text that follows to appear on the next page.
- Picture—Adds graphic images to the document.

- Title—Specifies a name for the template that appears in the print job list.

In addition, you may add static text in the print template.

The following print tags allow you to calculate additional information and add it to your reports:

- GetTotal—Calculates the total for a column of data.
- GetField—Gets the value of a field.
- GetCount—Calculates the number of records.

You can also calculate dates and times using the GetDate and GetTime tags, and you can retrieve the value of the registry using GetRegistry.

All of this information is stored in variables that you specify. In addition, you can assign any value to a variable using the SetVariable tag. You can present this information in your report using the Cell and EndofLine tags to create tables of information. See "Using Variables in Print Templates" on page 177 for an example of how variables can be used to present information in print templates.

Applet

Description

Specifies the applet to be printed.

Usage

Any data that appears in your document must be generated from the applets in the view. Therefore, you must create applets that will produce the data required for your document.

Use the Applet tag to specify any number of list or form applets within the template and to locate the applet in your document. In the form view, a maximum of two fields per row are supported.

Syntax

```
<TAG:Applet Name="APPLET NAME" Query="QUERY STRING" Caption="CAPTION"
Headerdivider="HEADERDIVIDER" Recorddivider="RECORDDIVIDER" Wrap="WRAP">
```

Parameter	Description
TAG:Applet	(Required) Applet tag indicator.
APPLET NAME	(Required) Applet name as defined in Siebel Tools.
QUERY STRING	(Optional) Specifies the query string directly set to the business component supporting the document. If the query string is not specified, the currently active query string in the view is applied to expose the relevant records.

Parameter	Description
CAPTION	(Optional) Specifies whether the captions in the form applets are suppressed. Valid values are TRUE (default) and FALSE. This parameter does not apply to list applets. Column headings in list applets are always enabled.
HEADERDIVIDER	(Optional) Specifies whether a divider is printed below the column heading in list applets. Valid values are TRUE and FALSE (default). This parameter does not apply to form applets.
RECORDDIVIDER	(Optional) Specifies whether a horizontal divider is printed between records in list applets. Valid values are TRUE and FALSE (default). This parameter does not apply to form applets.
WRAP	(Optional) Specifies whether text field values are word wrapped. Valid values are TRUE and FALSE (default). This parameter does not apply to numeric fields.

Example

<TAG:Applet Name="Order List" Headerdivider="TRUE">

Cell

Description

Prints the string stored in STRING or VARIABLE NAME at the specified location on the current line.

Usage

Use the Cell tag to create cells of data in a row in a table. Use it with the EndOfLine tag. If the data exceeds the width of the cell specified by Startat and Endat, it wraps to the next line.

Syntax

<TAG:Cell Startat="START" Endat="END" Name="STRING" Variable="VARIABLE NAME" Align=ALIGNMENT"

Parameter	Description
TAG:Cell	(Required) Required tag indicator.
START	(Required) Specifies the starting point of the cell, expressed as a percentage of the printable page width. For example, if START =10, then the cell starts at the point that is 10% from the left edge of the printable width.
END	(Required) Specifies the end point of the cell, expressed as a percentage of the printable page width. For example, if END=50, then the cell ends at the point that is 50% from the left edge of the printable width.
STRING	(Optional) String value. If Variable is not specified, then String is a required parameter.
VARIABLE NAME	(Optional) Name of variable where the value is stored. If String is not specified, then Variable is a required parameter.
ALIGNMENT	(Optional) Specifies the alignment of the string. Valid values are LEFT (default), CENTER, and RIGHT.

Example

```
<TAG:Divider Weight="3" Startat="20" Endat="80">
<TAG:Cell Startat="0" Endat="50" name="TOTAL:" Align="RIGHT">
<TAG:Cell Startat="50" Endat="100" Variable="vvv1total" Align="RIGHT">
<TAG:EndOfLine>
<TAG:Cell Startat="0" Endat="50" name="TOTAL Defective:" Align="RIGHT">
<TAG:Cell Startat="50" Endat="100" Variable="vvvDefectiveTotal" Align="RIGHT">
<TAG:EndOfLine>
<TAG:Cell Startat="0" Endat="50" name="TOTAL Good:" Align="RIGHT">
<TAG:Cell Startat="50" Endat="100" Variable="vvvGoodTotal" Align="RIGHT">
<TAG:EndOfLine>
<TAG:Cell Startat="0" Endat="50" name="Defective Count:" Align="RIGHT">
<TAG:Cell Startat="50" Endat="100" Variable="vvvDefectiveCount" Align="RIGHT">
<TAG:EndOfLine>
<TAG:Cell Startat="0" Endat="50" name="Good Count:" Align="RIGHT">
```

```
<TAG:Cell Startat="50" Endat="100" Variable="vvvGoodCount" Align="RIGHT">
<TAG:EndOfLine>
<TAG:Cell Startat="50" Endat="100" Variable="vvv3reg">
<TAG:EndOfLine>
<TAG:Cell Startat="50" Endat="100" Variable="vvv5str">
<TAG:EndOfLine>
<TAG:Cell Startat="0" Endat="30" Variable="vvv4date">
<TAG:Cell Startat="30" Endat="70" Variable="vvv4longdate">
<TAG:Cell Startat="70" Endat="100" Variable="vvv8time">
<TAG:EndOfLine>
```

Comment

Description

Adds code comments to the print template.

Usage

Use to add explanatory notes or documentation to explain and maintain the print template. Comment tags are ignored during printing and do not appear in the printed document.

Syntax

<REM: COMMENT>

Parameter	Description
REM	(Required) Comment tag indicator.
COMMENT	(Optional) Comment text.

Example

<REM: Quotation Print Template. This template includes four applets.>

Divider

Description

Renders a horizontal line across the width of the page.

Usage

Use to visually separate the different sections of your document. The Divider tag can be used between other tags; therefore, you may add a line between two applets. However, you cannot embed a Divider within an applet.

Syntax

<TAG:Divider Weight="WEIGHT" Startat="START" Endat="END">\

Parameter	Description
TAG:Divider	(Required) Divider tag indicator.
WEIGHT	(Optional) Specifies line thickness. Valid values are 1–5. The default value is 1, which produces a line 0.2 mm wide. Increasing the value by 1 increases the width by 0.2 mm. A 5 produces a line that is 1 mm wide.
START	(Optional) Percentage that specifies the starting point of the line. The default is 0%, the farthest left point.
END	(Optional) Percentage that specifies the end point of the line. The default is 100%, the farthest right point.

Example

<TAG:Divider Weight="2" Startat="20" Endat="80">

EndOfLine

Description

Specifies the end of a row and advances to the next line.

Usage

Use to create rows of data in a table. Used with the Cell tag.

Syntax

<TAG:EndOfLine>

Example

```
<TAG:Cell Startat="0" Endat="50" Name="TOTAL:" Align="RIGHT">
<TAG:Cell Startat="50" Endat="100" Variable="vvv1total" Align="RIGHT">
<TAG:EndOfLine>
```

Footer

Description

Adds footer information such as page numbers, the current date, and the current time.

Usage

You may, for example, always want the text "Thank you for your business!" to appear on the bottom of your receipts. If so, you may include this text in a Footer tag.

The Footer tag may be specified anywhere in the template file; however, it will always appear at the very bottom of your document. Only one Footer tag is applied to the document. If you have multiple Footer tags in the file, only the last Footer tag is used; any other Footer tags are ignored.

If you include a Footer tag in your template, then you must set the BottomMargin in setup.ini to a minimum of 20.

The footer has three sections, Left, Center, and Right, which are used to position the information in the footer.

The text in the footer is formatted in 10 point, Helvetica font. If Helvetica is not available on your system, the default font is used. You cannot customize text formatting of the Footer tag. The Format tag does not affect the formatting of the footer.

Syntax

<TAG:Footer Left="DATA" Center="DATA" Right="DATA">

Parameter	Description	
TAG:Footer	(Required) Footer tag indicator.	
DATA	(Optional) Static text or one of the following data tags:	
	Data tag	**Description**
	<page>	Page number
	<date>	Date of the printing
	<time>	Time of the printing

Example

<TAG:Footer Left="<date>" Right="<page>"

Format

Description

Specifies the characteristics of the text—the typeface or font, whether the text is in boldface, italics, or underlined, and the text alignment.

Usage

The font specification specifies the characteristics of all text that follows the tag until the next Format tag.

The text alignment tag (Align) behaves somewhat differently. When the applet is rendered, the field alignment specification takes precedence over the Align parameter.

Syntax

<TAG:Format Face="FONT FACE" Size="FONT SIZE" Bold="BOLD" Italic="ITALIC" Underline="UNDERLINE" Align="TEXT ALIGNMENT">

Parameter	Description
TAG:Format	(Required) Format tag indicator.
FONT FACE	(Required) Specifies the typeface of the text. Any typeface installed on the handheld device may be specified. There is no default for this parameter.
FONT SIZE	(Required) Specifies the text size. Any installed font size for an installed typeface on the handheld device may be specified. There is no default for this parameter.
BOLD	(Optional) Specifies whether the text is in bold. Valid values are TRUE or FALSE (default). If TRUE, text is set to bold.
ITALIC	(Optional) Specifies whether the text is in italic. Valid values are TRUE or FALSE (default). If TRUE, text is set to italic.
UNDERLINE	(Optional) Specifies whether the text is underlined. Valid values are TRUE or FALSE (default). If TRUE, text is set to underline.
TEXT ALIGNMENT	(Optional) Specifies the text alignment. Valid values are LEFT (default), RIGHT, and CENTER.

Example

<TAG:Format Face="Helvetica" Size="10">

GetCount

Description

Calculates the number of records in the selected rows and assigns this value to a variable.

Usage

Use GetCount to calculate the number of records and print this value in your report.

Syntax

<TAG:GetCount Appletname="APPLET NAME" Query="QUERY STRING" Variable="VARIABLE NAME">

Parameter	Description
TAG:GetCount	(Required) Required tag indicator.
APPLET NAME	(Required) Name of the applet.
QUERY STRING	(Optional) Search specification for the rows to be included in the row set. If not specified, the current search specification is used. The syntax for the query string is: Field Name|Query Statement. You may specify one or more query strings separated by a vertical bar (|). For example, Query="Name|LIKE 'Sieb*'"|"Location|LIKE 'H*'"
VARIABLE NAME	(Required) Name of variable where the value is stored.

Example

<TAG:GetCount Appletname="CS HH Product Bucket List Applet" Query="Status|Good" Variable="StatusGood">

GetDate

Description

Gets the current date, offsets the date by the number of days specified in Dayoffset, and stores the result in a variable.

Usage

Use GetDate to print a specific date in your report.

Syntax

<TAG:GetDate Dayoffset="NUMBER OF DAYS" Variable="VARIABLE NAME" Logformat="DATE FORMAT">

Parameter	Description
TAG:GetDate	(Required) Required tag indicator.
NUMBER OF DAYS	(Optional) Number of days by which the current date is offset. For example, –100 subtracts 100 days from the current date; 100 adds 100 days to the current date. The default is 0.
VARIABLE NAME	(Required) Name of variable where the value is stored.
DATE FORMAT	(Optional) TRUE prints the date in long date format (for example, Monday, February 11, 2002). FALSE prints the short date format (for example, 2/21/02). The default is FALSE.

Example

<TAG:GetDate Longformat="TRUE" Variable="LongDate">

GetField

Description

Gets the value of the specified column of the first row of selected rows and assigns the result to a variable.

Usage

Use GetField to print a value without developing a print applet. Design the query and sort strings so that the desired record is the first record in the selected rows. GetField only allows you to get the value in the first record.

Syntax

<TAG:GetField AppletName="APPLET NAME" Fieldname="FIELD NAME" Query="QUERY STRING" Sort="SORT STRING" Variable="VARIABLE NAME">

Parameter	Description
TAG:GetField	(Required) Required tag indicator.
APPLET NAME	(Required) Name of the applet.
FIELD NAME	(Required) Name of the field or column.
QUERY STRING	(Optional) Search specification for the rows to be included in the total. If not specified, the current search specification is used.

Parameter	Description
SORT STRING	(Optional) Sort specification. If not specified, the current sort specification is used. The syntax is: Field Name [ASC][DESC]. The default is ASC (Ascending). For example: Sort="Account DESC, Date". In the example, the Account field is sorted in descending order, then the date field is sorted in ascending order.
VARIABLE NAME	(Required) Name of variable where the value is stored.

Example

<TAG:GetField AppletName="CS HH Product Bucket List Applet" Fieldname="Quantity" Variable="ProdBucketListField">

GetRegistry

Description

Gets a value in the registry for Siebel Handheld and assigns it to a variable.

Usage

Use GetRegistry to print the contents of the registry key in your report.

Syntax

<TAG:GetRegistry Name="REGISTRY NAME" Variable="VARIABLE NAME">

Parameter	Description
TAG:GetRegistry	(Required) Required tag indicator.
REGISTRY NAME	(Required) Value the registry.
VARIABLE NAME	(Required) Name of variable where the value is stored.

Example

<TAG:GetRegistry Name="InstallDir" Variable="Registry">

GetTime

Description

Gets the current time, offsets the time by the number of seconds specified in Secondoffset, and assigns the result to a variable.

Usage

Use GetTime to print a specific time in your report.

Syntax

<TAG:GetTime Secondoffset="NUMBER OF SECONDS" Variable="VARIABLE NAME">

Parameter	Description
TAG:GetTime	(Required) Required tag indicator.
NUMBER OF SECONDS	(Optional)Number of seconds by which the current time is offset. For example, –100 subtracts 100 seconds from the current time; 100 adds 100 seconds to the current time. The default is 0.
VARIABLE NAME	(Required) Name of variable where the value is stored.

Example

<TAG:GetTime Secondoffset="3600" Variable="Time">

GetTotal

Description

Calculates the total for the specified column for all selected rows and assigns the value to a variable.

Usage

Use GetTotal to calculate a column total and display the total in your report.

Syntax

<TAG:GetTotal AppletName="APPLET NAME" FieldName="FIELD NAME" Query="QUERY STRING" Variable="VARIABLE NAME">

Parameter	Description
TAG:GetTotal	(Required) Required tag indicator.
APPLET NAME	(Required) Name of the applet.

Parameter	Description
FIELD NAME	(Required) Name of the field or column. Specify a field or column with numeric data. If you specify a field with non-numeric data, a 0 is assigned to the variable.
QUERY STRING	(Optional) Search specification for the rows to be included in the total. If not specified, the current search specification is used. The syntax for the query string is: Field Name|Query Statement. You may specify one or more query strings separated by a vertical bar (|). For example, Query="Name|LIKE 'Sieb*'"|"Location|LIKE 'H*'"
VARIABLE NAME	(Required) Name of variable where the value is stored.

Example

<TAG:GetTotal AppletName= "CS HH Product Bucket List Applet" Fieldname="Quantity" Variable="ProdBucketListTotal">

Header

Description

Adds header information such as page numbers, the current date, and the current time.

Usage

You may always want the current date and time to appear on your invoices. If so, you may include this information in a Header tag.

The Header tag may be specified anywhere in the template file; however, it always appears at the very top of your document. Only one Header tag is applied to the document. If you have multiple Header tags in the file, only the last Header tag is used; any other Header tags are ignored.

If you include a Header tag in your template, then you must set TopMargin in setup.ini to a minimum of 20.

The header has three sections, Left, Center, and Right, which are used to position the information in the header. The Left parameter aligns the text with the left margin, and the Right parameter right aligns the text with the right margin. The Center parameter centers the text in the header.

The text in the header is formatted in 10 point, Helvetica font. If Helvetica is not available on your system, the default font is used. You cannot customize text formatting of the Header tag. The Format tag does not affect the format of the header.

Syntax

<TAG:Header Left="DATA" Center="DATA" Right="DATA"

Parameter	Description	
TAG:Header	(Required) Header tag indicator.	
DATA	(Optional) Static text or one of the following data tags:	
	Data tag	**Description**
	<page>	Page number
	<date>	Date of the printing
	<time>	Time of the printing

Example

<TAG:Header Center="Company Confidential">

Page Break

Description

Forces text that follows the tag to appear on the next page.

Usage

Use this for multipage documents or forms.

Syntax

<TAG:PageBreak>

Parameter	Description
TAG:PageBreak	(Required) Page break tag indicator.

Example

<TAG:PageBreak>

Picture

Description

Specifies graphic files included in the print template.

Usage

Use to add graphic images, such as a company logo, to your documents.

The supported file formats are BMP and JPG.

Graphics files are located in the \Program Files\Siebel Handheld\templates directory on the handheld device. Specify the full path name of the file for the Name parameter.

Scaling or graphics operations, such as inverse image, are not supported.

The Alignment and Position parameters are used together to place the graphic. Alignment specifies the left edge, the center, or the right edge of the graphic. Position is distance, specified in millimeters, from the left edge of the paper.

- If Alignment ="Left" and Position="30", then the left edge of the graphic is set 30 millimeters from the left edge of the paper.
- If Alignment = "Center" and Position = "30", the center of the graphic is set 30 millimeters from the left edge of the paper.
- If Alignment = "Right" and Position = "30", the right edge of the graphic is set 30 millimeters from the left edge of the paper.

If Position is not specified, it defaults to 0. Depending on the Alignment setting, the graphic is left-aligned (Alignment = "Left"), the graphic is right-aligned (Alignment = "Right"), or the graphic is centered (Alignment ="Center")

Syntax

<TAG:Picture Name="FILE NAME" Alignment="ALIGNMENT" Position="POSITION">

Parameter	Description
TAG:Picture	(Required) Picture tag indicator.
FILE NAME	(Required) Specifies the full path and name of the graphics file.
ALIGNMENT	(Optional) Specifies the left edge, the center, or the right edge of the graphic. Used with the Position parameter to specify the horizontal position of the graphic. Valid values are LEFT (default), CENTER, and RIGHT.
POSITION	(Optional) Specified in millimeters, it is the distance from the left edge of the paper. Used with the Alignment parameter to specify the horizontal position of the graphic. The default is 0.

Example

<TAG:Picture Name="\Program Files\Siebel Handheld\templates\logo.bmp" Alignment="Right" Position="88">

SetVariable

Description
Sets a value to a variable.

Usage
Use SetVariable to assign a value to a variable. You can then print the value by specifying the variable in the Cell tag.

Syntax
<TAG:SetVariable Name="VALUE" Variable="VARIABLE NAME>

Parameter	Description
TAG:SetVariable	(Required) Required tag indicator.
VALUE	(Required) A value.
VARIABLE NAME	(Required) Name of variable where the value is stored.

Example
<TAG:SetVariable="100" Variable="TotalQuantity">

Static Text

You may add static text anywhere in your template by using the current Format settings. The text goes across the entire page width. If it exceeds the width of the page, it wraps to the next line.

Example
This quote is valid for 60 days from the date on this quotation.

Title

Description
Name used to identify the template.

Usage
The Title tag is a required tag in your print template file.

When you print your document, this title appears in the print job list. This title does not appear on the document that is printed. Use static text to add the document title to your template.

Syntax

<TAG:Title Title="TITLE">

Parameter	Description
TAG:Title	(Required) Title tag indicator.
TITLE	(Required) Title text. May be between 1–255 characters.

Example

<TAG:Title Title="Quotation">

```
<TAG:Title Title="Invoice">
<REM: Invoices Print Template. This template includes 3 applets.>
<TAG:Format Face="Helvetica" Size="16" Align="Center">
[Distributor Name
Street Address
City, Country Postal Code
Phone Number   Fax Number]
<TAG:Format Face="Helvetica" Size="20" Bold="TRUE" Align="Center">
INVOICE
<TAG:Format Face="Helvetica" Size="8" >
<TAG:Applet Name="SHCE Service Print Invoice Header Applet" Caption="TRUE" Wrap="TRUE">
<TAG:Format Face="Helvetica" Size="8">
Parts:
<TAG:Applet Name="SHCE Service Print Invoice Parts Applet" Maxlines="25" Line="FALSE"
Wrap="TRUE" Headerdivider="TRUE" Recorddivider="FALSE">
Time:
<TAG:Applet Name="SHCE Service Print Invoice Time Applet" Maxlines="25" Line="FALSE"
Wrap="TRUE" Headerdivider="TRUE" Recorddivider="FALSE">
Expenses:
<TAG:Applet Name="SHCE Service Print Invoice Expense Applet" Maxlines="25" Line="FALSE"
Wrap="TRUE" Headerdivider="TRUE" Recorddivider="FALSE">
<TAG:Divider Weight="2">
<TAG:Format Face="Helvetica" Size="12" Align="LEFT">
Customer
<TAG:Divider Weight="2">
<TAG:Format Face="Helvetica" Size="12" Align="LEFT">
Rep
```

Using Variables in Print Templates

The following example shows how to use variables in a print template.

```
<TAG:Title Title="Inventory Report">

<REM: **********Variable Assignment Examples**************>
```

```
<TAG:SetVariable name="THIS IS VARIABLE" variable="VarStr">
<TAG:GetRegistry name="InstallDir" variable="VarReg">
<TAG:GetDate dayoffset="10" variable="VarDate">
<TAG:GetDate longformat="TRUE" variable="VarLongDate">
<TAG:GetTime secondoffset="-3600" variable="VarTime">
<TAG:GetTotal appletname="CS HH Product Bucket List Applet" fieldname="Quantity"
variable="VarTotal">
<TAG:GetTotal appletname="CS HH Product Bucket List Applet" fieldname="Quantity"
query="Status|Good" variable="VarGoodTotal">
<TAG:GetTotal appletname="CS HH Product Bucket List Applet" fieldname="Quantity"
query="Status|Defective" variable="VarDefectiveTotal">
<TAG:GetField appletname="CS HH Product Bucket List Applet" fieldname="Quantity"
variable="VarQuantityField">
<TAG:GetCount appletname="CS HH Product Bucket List Applet" query="Status|Good"
variable="VarGoodCount">
<TAG:GetCount appletname="CS HH Product Bucket List Applet" query="Status|Defective"
variable="VarDefectiveCount">

<REM: **********Variable Assignment Examples End************>

<REM: Inventory Print Template. This template includes 3 applets. The ENU template>

<TAG:Format Face="Helvetica" Size="16" Align="Center">
[Distributor Name
Street Address
City, Country Postal Code
Phone Number   Fax Number]

<TAG:Format Face="Helvetica" Size="20" Bold="TRUE" Align="Center">
Inventory Report
<TAG:Format Face="Helvetica" Size="8" >
<TAG:Applet Name="CS HH Inv Loc Print Applet" Caption="TRUE" Wrap="TRUE">
<TAG:Applet Name="CS HH Product Bucket List Applet" Line="FALSE" Headerdivider="TRUE"
Recorddivider="FALSE" Wrap="TRUE">

<REM: *************Variable Usage Example***************>

<TAG:Divider Weight="3" startat="20" endat="80">
<TAG:Cell startat="0" endat="50" name="TOTAL:" Align="RIGHT">
<TAG:Cell startat="50" endat="100" variable="VarTotal" Align="RIGHT">
<TAG:EndOfLine>
<TAG:Cell startat="0" endat="50" name="TOTAL Defective:" Align="RIGHT">
<TAG:Cell startat="50" endat="100" variable="VarDefectiveTotal" Align="RIGHT">
<TAG:EndOfLine>

<TAG:Cell startat="0" endat="50" name="TOTAL Good:" Align="RIGHT">
<TAG:Cell startat="50" endat="100" variable="VarGoodTotal" Align="RIGHT"
<TAG:EndOfLine>

<TAG:Cell startat="0" endat="50" name="Defective Count:" Align="RIGHT">
<TAG:Cell startat="50" endat="100" variable="VarDefectiveCount" Align="RIGHT">
<TAG:EndOfLine>
```

```
<TAG:Cell startat="0" endat="50" name="Good Count:" Align="RIGHT">
<TAG:Cell startat="50" endat="100" variable="VarGoodCount" Align="RIGHT">
<TAG:EndOfLine>

<TAG:Cell startat="50" endat="100" variable="VarReg">
<TAG:EndOfLine>

<TAG:Cell startat="50" endat="100" variable="VarStr">
<TAG:EndOfLine>

<TAG:Cell startat="0" endat="30" variable="VarDate">
<TAG:Cell startat="30" endat="70" variable="VarLongDate">
<TAG:Cell startat="70" endat="100" variable="VarTime">
<TAG:EndOfLine>

<REM: ***********Variable Usage Example End***************>

<TAG:Divider Weight="2">
<TAG:Format Face="Helvetica" Size="12" Align="LEFT">
Warehouse Signature
<TAG:Divider Weight="2">
<TAG:Format Face="Helvetica" Size="12" Align="RIGHT">

Rep Signature
```

F Print Configuration Settings

This appendix includes the following topic:

Overview of Siebel Handheld Print Configuration Settings

The administrator sets up the default printer setting in the setup.ini file located in the setup directory.

There are a number of parameters to specify printing on the handheld. The parameters specify the following:

- Printer attributes (printer model, port, baud rate, and printer handshake).
- Printing attributes (portrait or landscape orientation, print density and quality, use of compression techniques, and advancing continuous–feed paper).
- Dimensions of the paper (U.S. or European standard paper sizes or custom–size paper).
- Margin settings of the pages (left, right, top, and bottom margins).

These parameters are specified in the [Printing] section of the setup.ini file located in the setup directory. The file contains default settings for all the printing parameters. Review the settings in the file to verify that these settings will work for your printing environment. If necessary, edit the settings as required. You may use any text editor to edit the file.

See Appendix E, "Print Tagging Language," for a list of the printing parameters and valid values.

The parameters are set using the following syntax:

PARAMETER_NAME = VALUE

For example: PrinterType = 1

The parameters are listed below in alphabetical order by the parameter name.

MarginBottom

Use the MarginBottom parameter to specify the dimensions of the bottom margin of your page. Specify the margin in millimeters using a whole number; decimal fractions are not valid. The default is 5 millimeters.

The data specified in the Footer tag of the print template is printed in the margin specified by MarginBottom. Therefore, if the print template includes a Footer tag, you must specify a wide enough bottom margin to accommodate the footer. MarginBottom should be set to a minimum of 20 millimeters. For more information, see "Footer" on page 167.

MarginLeft

Use the MarginLeft parameter to specify the width of the left margin of your page. Specify the margin in millimeters using a whole number; decimal fractions are not valid. The default is 5 millimeters.

MarginRight

Use the MarginRight parameter to specify the width of the right margin of your page. Specify the margin in millimeters using a whole number; decimal fractions are not valid. The default is 5 millimeters.

MarginTop

Use the MarginTop parameter to specify the dimensions of the top margin of your page. Specify the margin in millimeters using a whole number; decimal fractions are not valid. The default is 5 millimeters.

The data specified in the Header tag of the print template is printed in the margin specified by MarginTop. Therefore, if the print template includes a Header tag, you must specify a wide enough top margin to accommodate the footer. MarginTop should be set to a minimum of 20 millimeters. For more information, see "Header" on page 173.

PaperHeight

If the paper that your printer uses is not one of the supported standard sizes, then set PaperSize to 5 (custom paper dimensions). Then, specify the height of the paper, in millimeters, using the PaperHeight parameter. You must use a whole number; decimal fractions are not valid. The default is 280 millimeters. If you specify a value of 1–4 for PaperSize, the PaperHeight parameter is ignored.

Use the PaperWidth parameter to specify the width of the paper.

PaperSize

Specify one of the standard paper sizes or specify a custom paper size. (See Table 38.) If you specify custom paper size (5), you must also specify the PaperWidth and PaperHeight parameters. The default is custom paper dimensions.

See also PaperWidth and PaperHeight.

Table 38. Paper Dimensions

Value	Description
1	Letter size (8.5" x 11.5")
2	A4 (210 mm x 297 mm)
3	B5 (176 mm x 250 mm)
4	Legal size (8.5" x 14")
5 (default)	Custom paper dimensions (Specify PaperWidth and PaperHeight parameters.)

PaperWidth

If the paper that your printer is printing to is not one of the supported standard sizes, then set PaperSize to 5 (custom paper dimensions). Then, you must specify the width measurement of the paper, in millimeters, using the PaperWidth parameter. You must use a whole number; decimal fractions are not valid. The default is 5 millimeters. If you specify a value of 1–4 for PaperSize, the PaperWidth parameter is ignored.

Use the PaperHeight parameter to specify the length of the paper.

PrinterBaudrate

Use PrinterBaudrate to specify the speed of the data transmission for the printer. (See Table 39.)

Table 39. Printer Baud Rate

Value	Description
0	Always use 0 for LPT or Network printer ports or for COM ports that communicate with the printer at 4800 baud
1 (default)	9600 baud
2	19200 baud
3	38400 baud
4	57600 baud
5	115200 baud

PrinterCompressed

Use PrinterCompressed to specify the data compression mode. (See Table 40.) For the printers supported in this release, always set PrinterCompressed to 1.

Table 40. Data Compression

Value	Description
0	No compression techniques used
1 (default)	Use any known compression

PrinterDensity

Use the PrinterDensity parameter to get a lighter or darker output from the printer. Start with the default setting, 2, which corresponds to the printer manufacturer's default. As you get feedback from the field, you may need to adjust this setting.

The valid values are 0–4 where 0 is the lightest print density, 4 is the heaviest print density, and 2 is average print density.

PrinterDither

Use to specify the method by which different colors are represented. (See Table 41.) For the printers supported in this release, PrinterDither should always be set to 0.

Table 41. Print Dithering

Value	Description
0 (default)	Use color diffusion
1	Use dithering

PrinterDraftMode

Use PrinterDraftMode to specify the quality of the printing. (See Table 42.) For the printers supported in this release, PrinterDraftMode should always be set to 0.

Table 42. Print Quality

Value	Description
0 (default)	Use the highest quality printing
1	Use a lower quality printing if available

PrinterFormFeed

Use PrinterFormFeed to specify how paper advances to the next page. (See Table 43.) If using sheets of paper, set PrinterFormFeed to 0.

Specify 1 or 2 for continuous–feed paper:

- Set PrinterFormFeed to 1 if you do not need to conserve paper or if you want pages of uniform length. The printer scrolls to the length specified by the PageHeight parameter and trims the page there.
- Set PrinterFormFeed to 2 to conserve paper or if it does not matter if your pages are of varying length. In this case, the printer scrolls 25 millimeters beyond the last printed line and trims the page there. The scroll distance, 25 mm, is not customizable.

For both values, 1 and 2, the page length never exceeds PageHeight.

Table 43. Printer Form Feed

Value	Description
0 (default)	Printer sends a form feed command to the printer at the end of each page. Use this setting for sheet paper.
1	Printer scrolls the page to the length specified by the PaperHeight parameter.
2	Printer scrolls the page 25 millimeters past the last printed line of the page.

PrinterHandshake

Use PrinterHandshake to define the serial port handshake. (See Table 44.) The PrinterHandshake parameter applies only when the printer port is a Com1 or Com2 port. If PrinterPort is set to Com1 or Com2, then set PrinterHandshake to 1.

Table 44. Printer Handshake

Value	Description
0	Uses Xon/Xoff
1 (default)	Uses hardware control lines
2	No handshaking of serial port

PrinterOrientation

Specify whether the paper is to be oriented vertically (portrait) or horizontally (landscape). (See Table 45.) Not all printers can print in landscape orientation. Therefore, you will need to check the specifications for your printer.

Table 45. Paper Orientation

Value	Description
1 (default)	Portrait (vertical)
2	Landscape (horizontal)

PrinterPort

Specify the port to which the printer is connected with the PrinterPort parameter. (See Table 46.)

If PrinterPort is an LPT port or a network path (2 or 9), then the PrinterBaudrate must be set to 0.

NOTE: The communication port on the printer must be set to be compatible with the host PDA. For IrDA, the serial communication setting must be set correctly to allow reliable printing. You should contact your printer manufacturer and Field Software for more information.

Table 46. Printer Port

Value	Description
0	Com1: serial port
1	Com2: serial port
2	LPT: parallel port
3 (default)	IR Infrared port
4	Com3: serial port
5	Com4: serial port
6	Com5: serial port
7	Com6: serial port
8	Print to file OUTPUT.PRN
9	Print to network address
10	Bluetooth BlueCard solution by Wireless Solutions
11	Socket PNC Bluetooth card support
12	Bluetooth BlueCard solution by Anycom

Table 46. Printer Port

Value	Description
13	Com7: serial port
14	Com8: serial port

PrinterType

Specify the type of printer using the PrinterType parameter. See Table 47 for a list of printers supported by the Printer CE print driver used by the Siebel application.

Table 47. Supported Printers

Value	Description
0	Canon BubbleJet compatible printers
1	Citizen PD-04
2	Citizen PN60i
3	Epson ESC/P 2 compatible printers
4	Epson Stylus compatible printers
5	HP PCL compatible - includes DeskJet and LaserJet
6	Pentax PocketJet 200
7	Pentax PocketJet II
8	Seiko DPU-3445
9	Generic Epson compatible 180DPI 24-pin printers (LQ/ESC)
10	Generic Epson compatible 360DPI 24-pin printers (LQ/ESC)
11	Generic Epson compatible 203DPI 24-pin printers (LQ/ESC)
12	Extech 2"
13	Extech 3"
14	O'Neil MicroFlash
15	DymoCoStar label printers
16	Seiko LabelWriter printers
17	Extech 4"

Table 47. Supported Printers

Value	Description
18	SiPix A6 printer
102	Generic 40 column Dot Matrix Printer
108	Generic 80 column Dot Matrix Printer

Default Printing Settings

The following are the printing parameters from the setup.ini file. The values shown are the default settings.

```
[Printing]

PrinterType= 14
PrinterPort= 3
PrinterBaudrate= 1
PrinterHandshake= 1
PrinterOrientation= 1
PrinterDensity= 2
PrinterCompressed= 1
PrinterDither= 0
PrinterDraftMode= 0
PrinterFormFeed= 0
PaperSize= 5
PaperWidth= 115
PaperHeight= 280
MarginLeft= 5
MarginRight= 5
MarginTop= 5
MarginBottom= 5
```

Index

C

D

I

J

L

M

N

O

P

Q

R

S

T

U

V

W