



# **Siebel eSales Administration Guide**

Siebel Innovation Pack 2016

May 2016

**ORACLE®**

Copyright © 2005, 2016 Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

#### Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

#### Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

# Contents

## **Chapter 1: What's New in This Release**

## **Chapter 2: Siebel eSales Overview**

About Siebel eSales 11

The Siebel eSales Web Site 12

## **Chapter 3: Setting Up Siebel eSales**

Process of Setting Up Siebel eSales 13

Installing Siebel Business Applications 14

Setting Up Order Management for Siebel eSales 14

Customizing the eSales User Interface 14

    Creating Unstructured Content for the eSales Web Site 15

Creating Multilingual eSales Web Sites 15

Controlling User Access to the eSales Web Site 15

    Setting Up Default Responsibilities and User Types for eSales 16

    Defining Access Groups 17

    Setting Up Price Lists for Siebel eSales 18

Configuring Siebel eSales 19

Setting Up Personalization for Siebel eSales 19

Setting Up Shipping Charge Calculations for Siebel eSales 20

Encrypting Credit Card Data with RC2 Encryption 24

Customizing Email Notifications for Siebel eSales 25

Activating Workflows for Siebel eSales 26

## **Chapter 4: Integrating Siebel eSales with Third-Party Payment Applications**

About CyberSource Internet Commerce Suite 27

Installing CyberSource Software 27

Enabling Credit Card Transactions with CyberSource 28

- CyberSource Integration with Siebel eSales 28
  - Credit Card Transaction Service Business Service 29
  - Integration Objects 33
  - Data Maps 37
  - CyberSource Adapter Service Business Service 38
  - Address Verification with CyberSource 40
- Customizing Cybersource Integration 40
  - Modifying the Scripts in the CyberSource Adapter Service 40
  - Passing Data from Other Fields in the Siebel Database 41
  - Passing Data from Fields Not in the Siebel Database 41
  - Calling the Credit Card Transaction Service from Other Places in Siebel eSales 42
  - Calling the Credit Card Transaction Service from Other Business Components 42
  - Using Other Third-Party Payment Applications 42

## **Chapter 5: Integrating Siebel eSales with Third-Party Shipment Tracking Applications**

- About Shipment Tracking Applications 45
- Process of Setting Up Shipment Tracking 46
  - Setting the Shipment Tracking-Package Tracking User Properties 46
  - Verifying the External Shipment Tracking ASI User Properties 47
  - Configuring Web Services for Shipment Tracking 47
  - Extending Returned Data for Shipment Tracking 48
  - Changing the Package Sorting Order 49
  - Turning Off Package Sorting 49

## **Chapter 6: Administering Siebel eSales**

- Setting Quote Expiration Dates 51
- Managing Anonymous Shopping Carts 51
- About Setting Up Customer Accounts 52
  - About Allowing Accounts and Contacts to Use Purchase Orders 53
  - Allowing Contacts to Use Purchase Orders 54
- Modifying Credit Card Charges 54
  - Reversing Charges 55
  - Querying For Transactions With Error Messages 55

## **Chapter 7: Siebel eSales Shopping Cart**

- About eSales Shopping Cart 57
- Quote Types in Siebel eSales 57

|  |    |
|--|----|
| Shopping Service Business Service                            | 58 |
| Customizing the Shopping Service                             | 58 |
| Shopping Service Business Service Methods                    | 58 |
| Run-Time Events in Siebel eSales                             | 60 |
| Object Manager Support for Shopping Cart                     | 62 |
| Viewing Default Object Manager Parameters                    | 62 |
| Modifying Object Manager Parameters in the Web Client        | 63 |
| Modifying Object Manager Parameters in the Mobile Web Client | 64 |

## **Chapter 8: Siebel eSales Workflow Reference**

|  |    |
|--|----|
| About Siebel eSales Workflows              | 65 |
| Activating Siebel eSales Workflows         | 66 |
| Siebel eSales Quote and Template Workflows | 67 |
| eSales - Edit Quote Process                | 68 |
| eSales - Order Quote Process               | 70 |
| eSales - Reprice Process                   | 71 |
| eSales - Save Cart Process                 | 72 |
| eSales - Add To Template Process           | 73 |
| eSales - Delete Old Anonymous Quotes       | 74 |
| Siebel eSales Checkout Workflows           | 74 |
| SS Start Checkout Process                  | 76 |
| Credit Check - Quotes (eSales)             | 77 |
| eSales - Enter Shipping Details Process    | 78 |
| SS Save Shipping Address Details Process   | 80 |
| eSales - Enter Payment Details Process     | 82 |
| SS Save Payment Details Process            | 84 |
| eSales - Complete Checkout Process         | 86 |
| eSales - Authorize Order Process           | 87 |
| SS Authorize Order Process                 | 89 |
| eSales - Save Credit Card                  | 90 |
| SS Set Shipping Checkout Mode Process      | 91 |
| eSales - Save Shipping Address             | 92 |
| eSales - Save Account Shipping Address     | 92 |
| eSales - Set Primary Account Address       | 93 |
| eSales - Setting PO Status                 | 94 |
| eSales - Current User Type                 | 95 |
| SS Change Checkout Contact Detail Process  | 95 |
| SS Cancel Checkout Process                 | 96 |
| Siebel eSales Notification Workflows       | 96 |

|  |     |
|--|-----|
| Siebel eSales Purchase Approval Workflows              | 97  |
| eSales - Approve Order Process                         | 98  |
| eSales - Reject Order Process                          | 99  |
| Siebel eSales Order Modification Workflows             | 100 |
| eSales - Cancel Order Process                          | 100 |
| eSales - Modify Order Process                          | 101 |
| eSales - Reorder Process                               | 102 |
| Siebel eSales Error Message and View Display Workflows | 103 |
| eSales - Display Error Process                         | 103 |
| eSales - Login View                                    | 105 |
| eSales - Order Confirmation View                       | 106 |
| SS Order Confirmation View Process                     | 107 |
| eSales - Shopping Cart View                            | 108 |
| Auction Workflow                                       | 108 |

## **Appendix A: Shopping Service Methods Reference**

|   |     |
|---|-----|
| About the Shopping Service Business Service | 111 |
| Siebel eSales User Action Methods           | 111 |
| AddItem Method                              | 112 |
| AddToCart Method                            | 112 |
| AddToFavoriteList Method                    | 120 |
| SetShoppingCart Method                      | 122 |
| SetApplicationName Method                   | 124 |
| SetShoppingCartInfoChanged Method           | 125 |
| GetCartInfo Method                          | 125 |
| OrderTemplate Method                        | 126 |
| Siebel eSales Workflow Methods              | 128 |
| SaveCartAsTemplate Method                   | 129 |
| SaveCartAsQuote Method                      | 131 |
| DiscardCart Method                          | 133 |
| OrderQuote Method                           | 135 |
| ValidQuote Method                           | 136 |
| RemoveInvalidItems Method                   | 137 |
| CreateOrder Method                          | 138 |
| Reorder Method                              | 140 |
| GetShoppingCartStatus Method                | 142 |
| MoveToCart Method                           | 143 |
| RepriceQuote Method                         | 144 |
| SelectPrimary Method                        | 145 |
| SetBusCompFields Method                     | 146 |

|                          |     |
|--------------------------|-----|
| ValidatePayment Method   | 147 |
| ValidateAddress Method   | 148 |
| SetPendingRequest Method | 148 |

## Index





# 1

## What's New in This Release

### What's New in Siebel eSales Administration Guide, Siebel Innovation Pack 2016

Table 1 lists the changes in this revision of the documentation to support Siebel Innovation Pack 2016. The new features listed in this table are available only if you install and deploy Innovation Pack 2016. For more information, see *Siebel Patchset Installation Guides for Siebel Innovation Packs* (Doc ID 1614310.1) on My Oracle Support.

**NOTE:** Siebel Innovation Pack 2016 is a continuation of Siebel Innovation Pack 2015.

Table 1. New Product Features in Siebel eSales Administration Guide, Siebel Innovation Pack 2016

| Topic   | Description   |
|---|---|
| Implementing Siebel eSales Without Frames                             | Removed topic. References to frames were removed from the book because they are no longer available in the product.           |
| <a href="#">"Run-Time Events in Siebel eSales" on page 60</a>         | Modified topic. It describes the new Workflow_Checkout action set.  |
| <a href="#">"eSales - Edit Quote Process" on page 68</a>              | Modified topic. It describes how this updated process edits a saved quote.  |
| <a href="#">"eSales - Order Quote Process" on page 70</a>             | Modified topic. It describes how this updated process turns saved quotes into orders.   |
| <a href="#">"eSales - Save Cart Process" on page 72</a>               | Modified topic. It describes how this updated process saves a shopping cart.  |
| <a href="#">"SS Start Checkout Process" on page 76</a>                | New topic. It describes the workflow used to navigate a user to a single page checkout.                                       |
| <a href="#">"SS Save Shipping Address Details Process" on page 80</a> | New topic. It describes the workflow that validates shipping address details by associating it to a quote, and then saves it. |
| <a href="#">"SS Save Payment Details Process" on page 84</a>          | New topic. It describes the workflow used for selecting, entering, and modifying of payment information.                      |
| <a href="#">"SS Authorize Order Process" on page 89</a>               | New topic. It describes the workflow used to create a sales order for a specific quote.                                       |
| <a href="#">"SS Set Shipping Checkout Mode Process" on page 91</a>    | New topic. It describes the workflow that decides if an account's or contact's shipping address is displayed to the user.     |
| <a href="#">"eSales - Save Shipping Address" on page 92</a>           | Modified topic. It describes how this updated process saves shipping address information.                                     |
| <a href="#">"eSales - Save Account Shipping Address" on page 92</a>   | Modified topic. It describes how this updated process saves an account shipping address.                                      |

Table 1. New Product Features in Siebel eSales Administration Guide, Siebel Innovation Pack 2016

| Topic  | Description   |
|--|---|
| <a href="#">"SS Change Checkout Contact Detail Process" on page 95</a> | New topic. It describes the workflow that changes the contact details during checkout.  |
| <a href="#">"SS Cancel Checkout Process" on page 96</a>                | New topic. This topic replaces the eSales - Cancel Checkout topic. It describes the workflow that cancels the checkout process. |
| <a href="#">"eSales - Reorder Process" on page 102</a>                 | Modified topic. It describes how this updated process reorders a quote.   |
| <a href="#">"SS Order Confirmation View Process" on page 107</a>       | New topic. It describes the workflow that displays the Order Confirmation view.   |
| <a href="#">"SetShoppingCartInfoChanged Method" on page 125</a>        | New topic. It describes how the SetShoppingCartInfoChanged method checks the current status of the shopping cart.               |
| <a href="#">"GetCartInfo Method" on page 125</a>                       | New topic. It describes how the GetCartInfo method checks the number of root items in the shopping cart.                        |

## Additional Changes

This book also contains the following changes:

- Content in this guide was rewritten and reorganized to reflect the updated user interface.
- The following functionality is not currently available in Open UI:
  - Ability to save shopping cart as a template
  - Ability to save shopping cart as a favorite
  - Ability to do availability checks
  - Ability to transfer shopping cart

# 2

## Siebel eSales Overview

This chapter provides an overview of Oracle's Siebel eSales and the Siebel eSales Web site. It contains the following sections:

- ["About Siebel eSales" on page 11](#)
- ["The Siebel eSales Web Site" on page 12](#)

### About Siebel eSales

Siebel eSales allows your company to do business-to-business and business-to-consumer sales over the Web. It offers:

- Hierarchical multimedia product catalogs
- Product search
- User-specified product comparison
- Distributed catalog administration, allowing multiple-vendor catalogs
- Customer profiling and personalization, using Siebel Personalization
- Customizable order management processes, using Siebel Business Process Designer
- Online Shopping Carts, quotes, and orders, which include:
  - Order approval processes
  - Streamlined checkout processes
  - Integration with third-party taxation, credit checking, and payment processing applications
  - Integration with third-party fulfillment engines
- The following optional modules:
  - Siebel Configurator for Web-based product configuration
  - Siebel Pricer for one-to-one pricing and promotions using custom price lists
  - Eligibility for determining whether customers are eligible to purchase products.
  - Compatibility for defining global rules to determine which products are compatible.
  - Product promotions to create promotions that appear in the catalog.
  - Recommendations for defining rules to upsell and cross-sell products.

## The Siebel eSales Web Site

The home page gives customers an overview of the Web site and displays both the accessible catalogs and recommended products. Registered users can also access their account information.

The Web site uses the templates provided with Siebel customer applications. You could use these templates for creating portal pages to which you can add text, as well as links to HTML pages and documents. Portal pages can display marketing collateral, news articles, group news, financial information, press releases, and other kinds of unstructured content.

For information on customizing Web pages, see *Using Siebel Tools*.

Customers link to other pages by clicking tabs or hyperlinks on the home page. The following pages display more detailed information:

- **Catalogs.** This page provides access to the products and services on your company's Siebel eSales Web site.
- **Shopping Cart.** The Shopping Cart stores the products and services selected by customers. Each line item in the Shopping Cart displays the item's reference price and the customer's personalized price.
- **My Account.** The My Account page allows end users to update their user profiles, as well as view their saved quotes, saved orders, and registered products.

By registering and tracking their products, end users can receive product news or alerts, track warranty agreements, receive recommendations on scheduled services, and submit and track service requests related to their products.

Through the My Account page, end users can check their outstanding order status and reorder items.

Registered users with delegated customer administration privileges can add other users and edit user profiles. Delegated customer administrators can set the roles of other users within their accounts. The role indicates whether a registered user has both purchasing and administrative responsibilities.

All quotes and orders for a user are visible to that user on the My Account page, including those created in Siebel Sales and Siebel Call Center.

**NOTE:** If you do not want a user to be able to order a particular item from his or her favorites list or templates, you must delete the item from the appropriate price list or make it inactive. Removing the item from the catalog will not prevent the user from ordering that item.

If your company also has a license for Siebel Orders, depending on your configuration, you can automatically create asset records. If your company has also purchased Siebel eCustomer or Siebel Self Service, this does not prevent end users from registering products that they did not buy from your company. For more information about registered products, see *Siebel Self Service Administration Guide*.

# 3

## Setting Up Siebel eSales

This chapter describes how to set up Siebel eSales. It includes information about customizing the user interface, controlling user access, setting up responsibilities, and setting up price lists.

This chapter contains the following sections:

- ["Process of Setting Up Siebel eSales" on page 13](#)
- ["Installing Siebel Business Applications" on page 14](#)
- ["Setting Up Order Management for Siebel eSales" on page 14](#)
- ["Customizing the eSales User Interface" on page 14](#)
- ["Creating Multilingual eSales Web Sites" on page 15](#)
- ["Controlling User Access to the eSales Web Site" on page 15](#)
- ["Configuring Siebel eSales" on page 19](#)
- ["Setting Up Personalization for Siebel eSales" on page 19](#)
- ["Setting Up Shipping Charge Calculations for Siebel eSales" on page 20](#)
- ["Encrypting Credit Card Data with RC2 Encryption" on page 24](#)
- ["Customizing Email Notifications for Siebel eSales" on page 25](#)
- ["Activating Workflows for Siebel eSales" on page 26](#)

### Process of Setting Up Siebel eSales

To set up Siebel eSales, perform the following tasks:

- ["Installing Siebel Business Applications" on page 14.](#)
- ["Setting Up Order Management for Siebel eSales" on page 14](#)
- ["Customizing the eSales User Interface" on page 14.](#)
- ["Creating Multilingual eSales Web Sites" on page 15.](#)
- ["Controlling User Access to the eSales Web Site" on page 15.](#)
- ["Configuring Siebel eSales" on page 19.](#)
- ["Setting Up Personalization for Siebel eSales" on page 19.](#)
- ["Setting Up Shipping Charge Calculations for Siebel eSales" on page 20.](#)
- ["Encrypting Credit Card Data with RC2 Encryption" on page 24.](#)
- ["Customizing Email Notifications for Siebel eSales" on page 25.](#)
- ["Activating Workflows for Siebel eSales" on page 26.](#)

In addition to the tasks described in this chapter, you can also set up integration with third-party applications, as described in:

- Chapter 4, “Integrating Siebel eSales with Third-Party Payment Applications.”
- Chapter 5, “Integrating Siebel eSales with Third-Party Shipment Tracking Applications.”

## Installing Siebel Business Applications

To install and use Siebel eSales, you must have the following:

- **Siebel Web Engine and Related Components.** To deploy information on the Web, you must install the Siebel Web Engine and the components on which it depends. These include a Web server, Siebel Gateway Server, and Siebel Server.
- **Siebel Business Applications.** To perform administrative tasks such as server administration, creating catalogs, and managing Siebel Business Process Designer processes, you must install a Siebel employee application with the Siebel eSales license key provided.

For information about installing Siebel components, see *Siebel Installation Guide* for the operating system you are using. For information about installing Siebel clients, see the appropriate client installation guide.

## Setting Up Order Management for Siebel eSales

Siebel eSales allows customers to place their own orders. To set up Siebel eSales, you must perform many of the same tasks you perform to allow employees to place orders, such as defining products, creating catalogs, setting up shipping information, setting up tax calculations, and so on.

For more information, see the section about setting up order management in *Siebel Order Management Guide*.

## Customizing the eSales User Interface

Some of the modifications that you might want to perform to change the look and feel of your Siebel eSales Web site include the following:

- Modifying Web templates
- Changing colors
- Modifying the behavior caused by clicking a button
- Adding or removing applets
- Changing controls
- Making fields in forms required

For more information on customizing the eSales interface, see [“Creating Unstructured Content for the eSales Web Site” on page 15](#).

For more information on customizing the user interface, see *Using Siebel Tools*.

## Creating Unstructured Content for the eSales Web Site

Application administrators can create portal pages that contain unstructured content. These portal pages provide information such as announcements, press releases, and links to documents and HTML pages.

Siebel eSales includes one preconfigured portal page template, which is called Catalog Content View - News. This view contains four empty applets. You can modify this template to suit your company's business needs.

### *To access the Catalog Content View - News template*

- 1 Navigate to Administration Microsite > Page Administration.
- 2 In the Page list, select Web About Us.
- 3 Click the Web About Us hyperlink to access the page designer.

For more information about creating and displaying portal pages that include unstructured content, see *Siebel Applications Administration Guide*.

## Creating Multilingual eSales Web Sites

You can deploy Siebel eSales Web sites in more than one language. For information about deploying Siebel Business Applications in a multilingual environment, see *Siebel Global Deployment Guide*.

## Controlling User Access to the eSales Web Site

You might want to perform some of the following tasks to control user access in your Siebel eSales Web site.

- Using external authentication, such as LDAP
- Creating database users for LDAP to use in database access
- Defining visibility for the following:
  - Views (see [“Setting Up Default Responsibilities and User Types for eSales” on page 16](#))
  - Catalogs and categories (see [“Defining Access Groups” on page 17](#))
  - Products (see [“Setting Up Price Lists for Siebel eSales” on page 18](#))
- Assigning the proxy employee

- Creating users through registration or user administration
- Associating users with accounts
- Allowing these various levels of access:
  - Anonymous
  - Implicit login
  - Explicit login
- Customizing access to home and login pages
- Using single sign-on functionality
- Allowing another system to log in
- Adding fields to registration forms

For information on controlling user access, see *Siebel Security Guide*.

## Setting Up Default Responsibilities and User Types for eSales

Responsibilities control which views are visible to the user. Views, in turn, affect links and tabs. For example, if a user has a responsibility that does not permit access to My Company's Orders, no link appears in the My Account view that gives access to that view. If no views within a business object are visible, the screen tab that maps to that business object is not visible.

The responsibilities predefined for Siebel eSales are the following:

- **Web Anonymous User.** This responsibility grants view visibility to anonymous users. It lets users access views that do not have the Requires\_Explicit\_Login flag set to TRUE.
- **Web Registered User.** The end user of the application in a business-to-consumer model.
- **Web Corporate User.** The end user of the application in a business-to-business model. This responsibility is also the base responsibility for the Web Purchasing Manager and Web Delegated Customer Administrator.

Web Corporate Users are associated with an account and must be authorized by the Web Delegated Customer Administrator to access the site. A Siebel Administrator can also add new Web Corporate Users.

- **Web Purchasing Manager.** This is a power user in the buying organization. Web Purchasing Managers can approve or reject orders placed by Web Corporate Users. They can also view all company quotes and orders, as well as create, edit, and delete addresses in the corporate address book.

This responsibility has views only for access to company orders and quotes and company address book. Web Purchasing Managers must also have Web Corporate User responsibility to access Siebel eSales or Siebel eCustomer.



- **Web Delegated Customer Administrator.** This is a power user in the buying organization. Web Delegated Customer Administrators can add and remove Web Corporate Users, Web Purchasing Managers, and other Web Delegated Customer Administrators within your organization.

This responsibility has views for access to user administration only. Web Delegated Customer Administrators must also have Web Corporate User responsibility to access Siebel eSales or Siebel eCustomer. For purchase order approval and corporate address book privileges, they must also be given Web Purchasing Manager responsibility.

Web Purchasing Managers and Web Delegated Customer Administrators must have at least two responsibilities to see all of the views that they need, as described above. In Siebel eSales, workflow logic and other application logic can depend on the job role of the user, so where a user has more than one responsibility, a separate field called User Type must be set at the same time as the responsibilities. Workflows and other application logic use this field for decision-making in place of a responsibility. Set the user type for the highest level of view access needed for each job role, according to [Table 2](#).

Table 2. Responsibilities and User Type

| Job Role                             | Responsibilities   | User Type                            |
|--------------------------------------|--|--------------------------------------|
| Web Anonymous User                   | Web Anonymous User   | None                                 |
| Web Registered User                  | Web Registered User  | Web Registered User                  |
| Web Corporate User                   | Web Corporate User   | Web Corporate User                   |
| Web Purchasing Manager               | <ul style="list-style-type: none"> <li>■ Web Corporate User</li> <li>■ Web Purchasing Manager</li> </ul>               | Web Purchasing Manager               |
| Web Delegated Customer Administrator | <ul style="list-style-type: none"> <li>■ Web Corporate User</li> <li>■ Web Delegated Customer Administrator</li> </ul> | Web Delegated Customer Administrator |

**NOTE:** If you want to add a new responsibility that will be visible to Web Delegated Customer Administrators, you must set the Web Access field to Yes in the Responsibilities view of the Applications Administration screen. For more information about the Web Access feature, see *Siebel eService Administration Guide*.

For information on setting up responsibilities, see *Siebel Security Guide*.

## Defining Access Groups

You define access groups in the Access Groups view under Group Administration.

An access group is a collection of any combination of positions, organizations, divisions, accounts, and user lists. For example, an access group could consist of several business-to-business (B2B) customers for whom you want to grant access to a particular catalog.

The members of an access group are instances of party types other than Person. In other words, its members cannot be individual people. A user is associated with an access group by being associated with a position, organization, division, account, or user list that is a member of the access group.

Once you have defined access groups, you can associate them with catalogs and categories. Access groups determine which catalogs and categories are visible to the user. When displaying a list of products, for example, in the catalog, the user sees only those products that are in categories accessible to that user.

For information on defining access groups, see *Siebel Security Guide*.

## Setting Up Price Lists for Siebel eSales

Price lists set reference prices and promotional prices, and filter available products for a group or account.

When a customer logs in to the Siebel eSales Web site, the price list associated with the account is activated. The customer sees only products that are both on the activated price list and in catalogs and categories associated with the same access groups with which the customer is associated.

In a business-to-consumer (B2C) setting, a customer might not be associated with an account, or a customer's account might not have a default price list. Because you cannot identify customers browsing your Web site anonymously, you cannot associate those customers with an account. For these customers, you have to set a default price list that determines prices and product visibility.

For more information about price lists, see *Siebel Pricing Administration Guide*. For information on associating price lists with accounts for B2B customers, see the chapter on using accounts in Siebel Sales in *Siebel Applications Administration Guide*.

### Finding the Row ID of a Price List

You can find the row ID of a price list that you can use in setting it as the default price list.

#### *To find the row number of a price list*

- 1 In your Siebel employee application, navigate to Administration - Pricing > Price List.
- 2 Select the price list that you want to set as the default.
- 3 Click the menu button and select About Record.  
The About Record dialog box appears.
- 4 Record the row ID (Row #) of the price list.

### Setting the Default Price List

You must change the price list row ID supplied as seed data in your Siebel Business Application to the row ID of a price list that you have created.

**To set the default price list**

- 1 In your Siebel employee application, navigate to Administration - Data > List of Values.  
The List of Values appears.
- 2 Query on the Type field for PRICER\_MASTER\_PRICE\_LIST\_ID.
- 3 In the PRICER\_MASTER\_PRICE\_LIST\_ID record, enter the row number of the default price list in the Display Value field.

**NOTE:** You must update the existing PRICER\_MASTER\_PRICE\_LIST\_ID seed data row. Do not make the existing row inactive and then create a new one.

- 4 Click the Menu button, and then choose Clear Cache.

**NOTE:** You must clear the LOV cache for the new price list to take effect.

## Configuring Siebel eSales

You can use Siebel Tools to perform the following tasks to modify Siebel eSales to fit your business requirements:

- Configure Siebel business components, applets, views, and all other elements of a Siebel Business Application as needed, such as setting declarative properties or adding Web scripts.
- Create new controls, applets, views, or Web pages as needed. Add charts to views, if needed.
- Associate applets, views, and Web pages with the appropriate Web templates (.SWT files).
- Map the fields, controls, and applets to the placeholders in the appropriate template as follows:
  - Fields and controls are mapped into an applet.
  - Applets are mapped into a view.
  - Views are mapped into a Web page (a Web page container).
- Compile the repository changes into an SRF file.

For more information, see *Using Siebel Tools*.

## Setting Up Personalization for Siebel eSales

Siebel Personalization enhances your Siebel eSales Web site by showing information to customers that is specific to their profile, interests, and history. Personalized content includes greeting users by name, presenting them with content targeted to their needs and interests, showing them recommended products, and sending email messages addressed to them.

Siebel Personalization allows you to define rules to show and hide content dynamically during a user's experience with Siebel eSales.

Personalization deployment rules can depend on data such as users' personal profile information, date ranges, their company, products or services they have already purchased, or geographical information. These rules can also capture user data dynamically while the customer is browsing the Siebel eSales Web site.

Conditional expressions can be used to hide applets under certain conditions. The applets listed in [Table 3](#) are hidden depending on the application, the type of user, and whether or not the applets have any records in them.

Table 3. Applets That Can Be Hidden

| Applet                                   | View                     | Effect of Conditional Expression   |
|--|--------------------------|--|
| My Company Applet (eSales)               | My Account               | Hides corporate purchase information from customers who are not B2B power users (that is, do not have Web Purchasing Manager or Web Delegated Customer Administrator responsibility) |
| Quote Empty Form Applet (eSales)         | Shopping Cart            | Shows an Empty form when the Shopping Cart is empty  |
| Quote Form Applet - Current (eSales)     | Shopping Cart            | Hides the Shopping Cart header form when the Shopping Cart is empty  |
| Quote Item List Applet (eSales)          | Shopping Cart            | Hides the Shopping Cart when empty   |
| Recommended Product List Applet (eSales) | Home Page, Shopping Cart | Hides the Recommended Products applet from Siebel eEvents.   |

Personalization is principally managed using the Personalization Administration screen in your Siebel Business Application. For information about administering personalization, see *Siebel Personalization Administration Guide*.

## Setting Up Shipping Charge Calculations for Siebel eSales

Shipping charges are calculated for quotes as part of the checkout process of Siebel eSales. Siebel eScripts are used for the calculations and have been provided as templates. You modify the scripts to reflect your shipping processes.

## Shipping Calculation Process Workflow

The Shipping Calculation Process workflow is shown in [Figure 1](#).

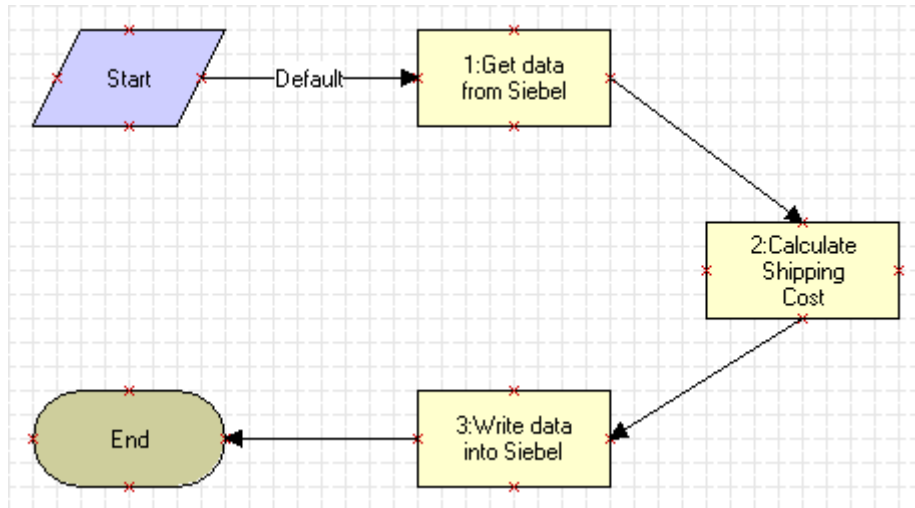


Figure 1. Shipping Calculation Process

When the Shipping Calculation Process workflow is called by the eSales - Complete Checkout Process workflow ([“eSales - Complete Checkout Process”](#) on page 86), the following events happen:

- The EAI Siebel Adapter business service passes Siebel data to the Shipping Cost Service (eScript) business service, using the business service method argument SiebelMessage.
- The Siebel eScripts in the Shipping Cost Service business service calculate the shipping charges based on the cost, the carrier, and the shipping method.
- The EAI Siebel Adapter business service passes the results to the Siebel database, using the business service method argument SiebelMessage.

## Business Services

Two business services are used as part of the Shipping Calculation Process workflow, one of them twice:

- EAI Siebel Adapter, in step 1: Get data from Siebel and step 3: Write data into Siebel

- Shipping Cost Service (eScript), in step 2: Calculate Shipping Cost

The call to the Shipping Cost Service (eScript) business service sets the business service method argument values shown in [Table 4](#).

Table 4. Business Service Method Argument Values

| Business Service Method Argument | Value   |
|----------------------------------|---|
| MapName                          | CalculateShippingCost   |
| SiebelMessage                    | SiebelMessage is what is passed back and forth between the business service and the workflow. It contains the input and output information in a structured format.  |
| Input Integration Object Name    | If you want to use this business service for another business component, you must create a new input integration object and define its value here. The value should match the equivalent value for the business service in the workflow.  |
| Output Integration Object Name   | If you want to use this business service for another business component, you must create a new output integration object and define its value here. The value should match the equivalent value for the business service in the workflow. |

## Integration Objects

Integration objects are used to pass static or dynamic values to business services. The architecture is modular in that you can create and use different integration objects for the same core business service.

Shipping charge calculation uses the following integration objects:

- **ShippingCost - Quote (Siebel Output)**. Its integration components are shown in [Table 5](#).

Table 5. ShippingCost - Quote (Siebel Output) Integration Components

| Integration Component | Fields                            |
|-----------------------|-----------------------------------|
| Quote                 | Account Id                        |
|                       | Carrier Priority                  |
|                       | Currency Code                     |
|                       | Current Quote Ship To City        |
|                       | Current Quote Ship To Country     |
|                       | Current Quote Ship To Postal Code |
|                       | Current Quote Ship To State       |
|                       | Quote Number                      |
|                       | Shipping Method                   |
|                       | Quote Item                        |
| Part Number           |                                   |
| Quantity              |                                   |
| Sales Product Flag    |                                   |

- **ShippingCost - Quote (Siebel Input)**. Its integration component is Quote, with the following fields:
  - Freight
  - Quote Number

For more information on integration objects, see *Integration Platform Technologies: Siebel Enterprise Application Integration and Using Siebel Tools*.

## Business Service Scripts

The following scripts are used in the Shipping Cost Service (eScript) business service:

- **(declarations)**. This script sets the following values—Lower Ranges, Upper Ranges, Base Costs, and Rates. It associates the Type of carrier, Priority of the service, and Cost of the service.
- **CalculateShippingCost**. This script calculates the shipping cost based upon the information passed from the quote and the other scripts in the business service.
- **GetShippingMethodCost**. This script determines the shipping method and shipping carrier costs to use.
- **GetShippingRate**. This script determines what shipping rate to use.

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

## Modifying Shipping Charge Calculations

You customize shipping charge calculations in Siebel Tools by modifying the scripts that make up the Shipping Cost Service (eScript) business service.

For more information on Siebel eScripts, see *Siebel Tools Online Help*.

### To modify shipping charge calculations

- 1 In Siebel Tools, lock the EAI Credit Card project.
- 2 Select the Business Service object.
- 3 Query for Ship\*.
  - The Shipping Cost Service (eScript) business service appears.
- 4 Right-click Shipping Cost Service (eScript), and then choose Edit Server Scripts.
  - The server script editor appears.
- 5 Edit the scripts, and then close the edit window.
- 6 Compile the .srf file, selecting the Locked projects radio button.

# Encrypting Credit Card Data with RC2 Encryption

Credit card information is encrypted with RC2 encryption technology from RSA. The encrypted fields used in Siebel eSales are shown in [Table 6](#).

**CAUTION:** Make sure that you back up the key file regularly. If the key file is lost, you cannot retrieve credit card data.

Table 6. Encrypted Fields in Siebel eSales

| Business Component       | Field              | Table          | Column       |
|--------------------------|--------------------|----------------|--------------|
| Personal Payment Profile | Account Number     | S_PER_PAY_PRFL | PAY_ACCT_NUM |
| Quote                    | Credit Card Number | S_DOC_ORDER    | CC_NUMBER    |
| Order Entry - Orders     | Credit Card Number | S_SRC_PAYMENTS | CC_NUM       |

**NOTE:** If you are upgrading to the current version of Siebel Business Applications from version 7.0.x or earlier, you must convert Oracle’s proprietary encryption for these fields to RC2 encryption. To do this you must run an encryption upgrade script after the main upgrade script but before the Unicode upgrade.

For general information about RSA encryption, see *Siebel Security Guide*. For more information about running the upgrade script, see *Siebel Database Upgrade Guide* for the operating system you are using.



# Customizing Email Notifications for Siebel eSales

Siebel eSales sends an email notification to a user when the status of that user's order is changed to Confirmed or Shipped. Notifications of other events are also sent—for example, when the orders of business-to-business customers are approved or rejected.

The tasks involved in customizing notifications for Siebel eSales are the following:

- Setting up communications driver profiles. See [“Setting Up Communications Driver Profiles” on page 25](#).
- Modifying email workflows. See [“Modifying Email Workflows” on page 25](#).
- Modifying email templates. See [“Modifying Email Templates” on page 25](#).

For information on creating notifications, see *Siebel Communications Server Administration Guide* and *Siebel Applications Administration Guide*.

## Setting Up Communications Driver Profiles

Communications drivers send messages. They are accessed using communications driver profiles, which set parameters that can override those of the driver.

Communications drivers are part of the Siebel eSales application. Profiles are created in the Communications Drivers and Profiles view under the Communications Administration screen.

For information about setting up communications driver profiles, see *Siebel Communications Server Administration Guide*.

## Modifying Email Workflows

Workflows control notifications in Siebel eSales. You can modify existing workflows and create new ones.

For more information about Siebel eSales workflows for email notifications, see [“Siebel eSales Notification Workflows” on page 96](#). For more information about defining rules to enable automatic email notification, see *Siebel Business Process Designer Administration Guide*.

## Modifying Email Templates

Templates provide the content for automated messages, such as confirmations of orders being placed and shipped.

### *To modify an email template*

- 1 In your Siebel employee application, navigate to Administration - Communications > All Templates.  
The Templates view appears.
- 2 Select a template—for example, Order Confirmation.

- 3 Make modifications to the template record's Text field.  
You can also copy and paste available substitutions, such as [Street Address].
- 4 Click the Template Items tab to attach preformatted files with template content, such as a standard header or footer.

## Activating Workflows for Siebel eSales

Siebel eSales uses multiple workflow processes to implement its features. Those workflows that are specific to eSales are listed in [Chapter 8, "Siebel eSales Workflow Reference."](#) Before using Siebel eSales, you must activate these workflows to make the features available. For information about activating workflows, see *Siebel Business Process Designer Administration Guide*.

# 4

## Integrating Siebel eSales with Third-Party Payment Applications

This chapter describes how to set up and use CyberSource, a third-party payment application that is supported by Siebel eSales. To use the CyberSource application, you must buy a separate license from CyberSource. You can also use other third-party applications for payment integration.

This chapter contains the following sections:

- “About CyberSource Internet Commerce Suite” on page 27
- “Installing CyberSource Software” on page 27
- “Enabling Credit Card Transactions with CyberSource” on page 28
- “CyberSource Integration with Siebel eSales” on page 28
- “Customizing Cybersource Integration” on page 40

### About CyberSource Internet Commerce Suite

Siebel eSales retrieves credit card information from CyberSource as part of the eSales - Authorize Order Process workflow (Figure 15 on page 87) when the customer clicks Confirm Order on the Order Summary page.

The CyberSource Internet Commerce Suite (ICS) is a set of third-party eCommerce applications. Siebel eSales uses the ICS to authorize credit card payments.

Siebel eSales accesses the ICS using an application programming interface (API), which is installed on the Web server. Messages are sent using CyberSource’s Simple Commerce Messaging Protocol (SCMP), which is based on Secure/Multipurpose Internet Mail Extensions (S/MIME), for authentication and encryption.

These messages contain a set of fields that describe the ICS application requests and provide information about the customer, the merchant, and the order form. SCMP messages are digitally signed and armored for transmission over an HTTP connection, allowing them to pass through firewalls and proxy servers.

For more information on the ICS, see *Internet Commerce Suite Version 2 Application Programming Interface (API) Developer’s Guide*, which is available from the CyberSource Support Center (<http://www.cybersource.com/support>).

**NOTE:** You must obtain a merchant ID from CyberSource to access the Support Center.

### Installing CyberSource Software

The steps to take for installing CyberSource are as follows:

- 1 Obtain a merchant ID.
- 2 Download the compressed file for the CyberSource Developer Kit (CDK) for your platform.
- 3 Install the CDK on the Siebel Server; set the system variables for the CDK and libraries.

For instructions on installing CyberSource applications, see the CyberSource documentation. For information on which versions of CyberSource software are supported by Siebel Business Applications, see *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.

It is recommended that you download and use the latest version of CyberSource software from <http://www.cybersource.com>.

## Enabling Credit Card Transactions with CyberSource

To enable credit card transactions, you must set Siebel Server object manager parameters.

### *To enable credit card transactions with CyberSource*

- 1 Navigate to Server Administration > Components > Component Parameters.
- 2 Set the parameters for the eSales Object Manager as shown in the table that follows.

| Parameter     | Value              |
|---------------|--------------------|
| CCAEnable     | TRUE               |
| CCAMerchantId | <i>merchant_id</i> |
| CCAServerHost | ics2test.ic3.com   |
| CCAType       | CyberSourceICS2    |

## CyberSource Integration with Siebel eSales

Integration of Siebel eSales with third-party payment applications involves three business services:

- Credit Card Transaction Service (Universal Credit Card Service)
- CyberSource Adapter Service
- Shopping Service

This section contains the following topics:

- [“Credit Card Transaction Service Business Service” on page 29](#)
- [“Integration Objects” on page 33](#)
- [“Data Maps” on page 37](#)

- [“CyberSource Adapter Service Business Service” on page 38](#)
- [“Address Verification with CyberSource” on page 40](#)

## Credit Card Transaction Service Business Service

The Credit Card Transaction Service process flow follows.

- Siebel EAI passes data from a quote record and the method name to a Siebel output integration object.
- The Siebel Data Transformation Engine maps the Siebel output integration object to a CyberSource input integration object.
- The Siebel EAI XML Converter converts the vendor input integration object into XML, using the CC XML Converter Business Service, and passes the XML data and method name to the CyberSource Adapter Service.
- The CyberSource Adapter Service business service processes the data, converts it into Siebel eScript, and passes it to CyberSource using the CyberSource API.
- The CyberSource Adapter Service gets the payment information from CyberSource in Siebel eScript through the CyberSource API.
- The CyberSource Adapter Service formats the payment information into an XML string.
- The Siebel EAI XML Converter converts the XML into a vendor output integration object.
- The Siebel Data Transformation Engine maps the vendor output integration object to a Siebel input integration object.
- Siebel EAI updates the quote record with the payment information.

### Business Service Methods

The Credit Card Transaction Service has business service methods for each of the following types of credit card transaction:

- **AuthCharge.** This transaction authorizes and settles charges. It makes sure that the funds are available, reserves them, and requests that they be transferred in one call. It does this in the case of immediate order fulfillment (for example, digital downloads).
- **Authorization.** Called from the eSales - Authorize Order Process workflow, this transaction allocates to the merchant a specific amount of the credit available to the cardholder, typically for one week. No funds are transferred.

The merchant authorizes a credit card to make sure of the following:

- The customer has the funds to pay for the purchase.
- These funds are reserved until the merchant is able to ship the product. Credit card companies typically state that a merchant should not settle until the product has been shipped.

- **Charge.** This transaction is used to settle charges. A settlement request includes the authorization code and requests that funds be transferred from the cardholder's account to the merchant's bank account.
- **Process.** This transaction is not used, but is available to customers for calling other methods from the third-party adapter.
- **Refund.** A refund takes place after a settlement when the merchant needs to transfer money back to the cardholder, as in the case of a return.
- **Reverse.** This transaction releases the amount of money reserved by an authorization. Although it is not supported by CyberSource, Reverse is important because authorizations reserve part of a cardholder's credit and therefore reduce the total amount of credit available.

The arguments for these methods are:

- **Input:**
  - Row Id
  - Siebel Input Integration Object
  - Siebel Output Integration Object
  - Siebel to Vendor Map Name
  - Vendor to Siebel Map Name
  - Vendor Name
- **Output:**
  - Credit Card Response Code
  - Credit Card Response Message
- **Validate.** Validation checks the credit card number, using the Mod 10 algorithm to make sure that it is valid, and the expiration date. The Siebel Business Application—not CyberSource—does this during rapid checkout and after selecting and saving payment information.

The arguments for Validate are:

- **Input:**
  - Credit Card Number
  - Credit Card Type
  - Expiration Month
  - Expiration Year
  - Return Error Code
- **Output:** Error Message

## Business Service User Properties

The business service user properties specify the adapter business service to call, the configuration object to use, the vendor integration objects to use, and the text of the error messages. The active business service user properties are shown in [Table 7](#).

Table 7. Credit Card Transaction Service User Properties

| Business Service User Property                       | Value  |
|--|--|
| Create Order on Web                                  | 1, 18, 19, 20, 21<br><br>These response codes will not return an error. Administrators should query for orders with these message types and investigate them manually. For more information, see <a href="#">Chapter 6, "Administering Siebel eSales."</a> |
| CyberSourceICS2:Adapter Business Service             | CyberSource Adapter Service  |
| CyberSourceICS2:AuthCharge Object (Vendor Input)     | CreditCard - Authorization (CyberSource Input)   |
| CyberSourceICS2:AuthCharge Object (Vendor Output)    | CreditCard - AuthCharge (CyberSource Output)   |
| CyberSourceICS2:Authorization Object (Vendor Input)  | CreditCard - Authorization (CyberSource Input)   |
| CyberSourceICS2:Authorization Object (Vendor Output) | CreditCard - Authorization (CyberSource Output)  |
| CyberSourceICS2:Charge Object (Vendor Input)         | CreditCard - Charge (CyberSource Input)  |
| CyberSourceICS2:Charge Object (Vendor Output)        | CreditCard - Charge (CyberSource Output)   |
| CyberSourceICS2:Configuration Object                 | CreditCard - Configuration (CyberSource)   |
| CyberSourceICS2:Refund Object (Vendor Input)         | CreditCard - Refund (CyberSource Input)  |
| CyberSourceICS2:Refund Object (Vendor Output)        | CreditCard - Refund (CyberSource Output)   |
| Response Message: 1                                  | Transaction was successful.  |
| Response Message: 2                                  | Transaction was not successful. Please call your credit card company or try another credit card.   |
| Response Message: 3                                  | Transaction was declined. Please select another credit card.   |
| Response Message: 4                                  | There is a problem with this credit card. Please select another credit card.   |

Table 7. Credit Card Transaction Service User Properties

| Business Service User Property | Value   |
|--------------------------------|---|
| Response Message: 5            | Your credit card has expired. Please select another card.   |
| Response Message: 6            | Please check the credit card details or select another card.  |
| Response Message: 7            | Your credit card is invalid. Please check the card details or select another credit card.   |
| Response Message: 8            | Transaction declined by the bank. Please select another credit card.  |
| Response Message: 9            | Your credit card number does not match the card type. Please check the card details or select another credit card.  |
| Response Message: 10           | The billing address you entered does not match the billing address of your credit card. Please check the card details or select another card.<br><br>This response is given when CyberSource returns DAVSNO. For more information, see <a href="#">"Address Verification with CyberSource" on page 40</a> . |
| Response Message: 11           | Please check to make sure you have entered complete credit card information.  |
| Response Message: 12           | There is not enough available credit on this card. Please select another credit card.   |
| Response Message: 13           | There was an error in processing. Please check the credit card information and try again or select another credit card.   |
| Response Message: 14           | This type of credit card is not accepted. Please select another card.   |
| Response Message: 15           | There is no corresponding, unused authorization record. Please authorize for this amount prior to settling.   |
| Response Message: 16           | A credit or refund is only allowed when there is a corresponding settlement transaction referenced.   |
| Response Message: 17           | Transaction has been settled, so it cannot be voided.   |
| Response Message: 18           | You are not authorized to perform this transaction type.  |
| Response Message: 19           | Error happened in adapter business service. Please contact your application administrator.  |
| Response Message: 20           | Error occurred in CyberSource. Please contact your application administrator.   |
| Response Message: 21           | The connection to CyberSource has timed out. Please try the operation later.  |



The complete response messages, and not the codes, are shown in the Siebel eSales application. For more information on modifying business service user properties, see *Using Siebel Tools*.

## Integration Objects

Integration objects pass static or dynamic values to the business service. The architecture is modular in that you can create and use different integration objects for the same core business service.

The integration objects for the Quote and Order business components and CyberSource are detailed in this section.

## Configuration

CreditCard - Configuration (CyberSource) has the user properties shown in [Table 8](#).

Table 8. Credit Card - Configuration (CyberSource) User Properties

| User Property     | Value              | Comments  |
|-------------------|--------------------|---|
| VendorDII         | ics2api            | Do not use the prefix and extension.<br>This is necessary for compatibility with Solaris.   |
| customer_email    | nobody@nowhere.com | CyberSource enforces the format.  |
| customer_phone    | (000) 000-0000     | CyberSource requires at least six digits.   |
| ignore_avs        | no                 | CyberSource needs "yes" or "no."<br>This is only used for the AuthCharge transaction.   |
| merchant_id       | ICS2Test           | CyberSource varchar(20), case-sensitive.<br>Change this to your merchant ID. You can obtain a merchant ID for each division of your organization. The merchant IDs are set in the Divisions view under the Group Administration screen. |
| server_host       | ics2test.ic3.com   |   |
| server_port       | 80                 |   |
| timeout           | 90                 | In seconds. The default is 90.  |
| decline_avs_flags | N                  | Comma-separated list of AVS codes that result in an AVS decline.<br>For more information, see <a href="#">"Address Verification with CyberSource" on page 40</a> .  |

## Authorization

- **CreditCard - Authorization (CyberSource Input).** Its integration component is Invoice, with the following fields:

- |                    |                          |                            |
|--------------------|--------------------------|----------------------------|
| ■ Amount           | ■ Bill To City           | ■ Bill To Country          |
| ■ Bill To State    | ■ Bill To Street Address | ■ Bill To Street Address 2 |
| ■ Bill To Zipcode  | ■ Card Expiration Month  | ■ Card Expiration Year     |
| ■ Card Holder Name | ■ Card Issue Number      | ■ Card Number              |
| ■ Card Start Month | ■ Card Start Year        | ■ Card Verification Value  |
| ■ Currency Code    | ■ Duty Amount            | ■ Email Address            |
| ■ Ignore AVS       | ■ Ignore Bad CV          | ■ Merchant Id              |
| ■ National Tax     | ■ Phone Number           | ■ Product Code             |
| ■ Purchaser Code   | ■ Purchaser VAT Reg No   | ■ Quantity                 |
| ■ Row Id           | ■ Ship From Zipcode      | ■ Ship To Country          |
| ■ Ship To Zipcode  | ■ Summary Commodity Code | ■ Tax Amount               |
| ■ Tax Indicator    | ■ VAT rate               |                            |

- **CreditCard - Authorization (CyberSource Output).** Its integration component is Invoice, with the following fields:

- |                           |                           |
|---------------------------|---------------------------|
| ■ AVS                     | ■ Application Resp Code   |
| ■ Application Resp Flag   | ■ Application Resp Msg    |
| ■ Authorization Code      | ■ Authorization Resp Code |
| ■ Authorization Resp Flag | ■ Authorization Resp Msg  |
| ■ Authorization Response  | ■ Authorized Amount       |
| ■ Authorized Time         | ■ CV Result               |
| ■ Request Id              | ■ Row Id                  |

- **CreditCard - Payments.** Its integration component is Credit Card Payments, with the following fields:

- |                                  |                                |
|----------------------------------|--------------------------------|
| ■ Authorization Code             | ■ Bill To City                 |
| ■ Bill To Country                | ■ Bill To State                |
| ■ Bill To Street Address         | ■ Bill To Street Address 2     |
| ■ Bill To Zipcode                | ■ Card Holder Name             |
| ■ Card Number                    | ■ Card Type                    |
| ■ Card Verification Number       | ■ Credit Card Response Code    |
| ■ Credit Card Transaction Amount | ■ Credit Card Transaction ID   |
| ■ Credit Card Transaction Status | ■ Credit Card Transaction Time |
| ■ Credit Status                  | ■ Credit Status Message        |
| ■ Currency Code                  | ■ Expiration Month             |
| ■ Expiration Year                | ■ Merchant Id                  |

- Row Id
- Transaction Id

- **CreditCard - Quote - Authorization (Siebel Input)**. Its integration component is Quote, with the following fields:

- Credit Card Authorization Code
- Credit Card Transaction Amount
- Credit Card Transaction ID
- Credit Card Transaction Status
- Quote Number

- **CreditCard - Quote - Authorization (Siebel Output)**. Its integration component is Quote, with the following fields:

- Amount
- Bill To Country
- Bill To Street Address
- Bill To Zipcode
- Card Expiration Year
- Card Number
- Currency Code
- Home Phone
- Work Phone
- Bill To City
- Bill To State
- Bill To Street Address 2
- Card Expiration Month
- Card Holder Name
- Card Type
- Email Address
- Quote Number

## Settle (Charge)

- **CreditCard - Charge (CyberSource Input)**. Its integration component is Invoice, with the following fields:

- Amount
- Authorization Code
- Currency Code
- Merchant Descriptor
- Merchant Id
- Product Code
- Purchaser Code
- Quantity
- Ship From Zipcode
- Ship To Zipcode
- Tax Amount
- VAT rate
- Auth Request Id
- Authorization Type
- Duty Amount
- Merchant Descriptor Contact
- National Tax
- Product Name
- Purchaser VAT Reg No
- Row Id
- Ship To Country
- Summary Commodity Code
- Tax Indicator

- **CreditCard - Charge (CyberSource Output)**. Its integration component is Invoice, with the following fields:

- |                           |                         |
|---------------------------|-------------------------|
| ■ Application Resp Code   | ■ Application Resp Flag |
| ■ Application Resp Msg    | ■ Charge Resp Code      |
| ■ Charge Resp Flag        | ■ Charge Resp Msg       |
| ■ Charge Trans Ref Number | ■ Charged Amount        |
| ■ Charged Time            | ■ Request Id            |
| ■ Row Id                  |                         |

### Authorize and Settle (AuthCharge)

- **CreditCard - AuthCharge (CyberSource Output)**. Its integration component is Invoice, with the following fields:

- |                           |                           |
|---------------------------|---------------------------|
| ■ AVS                     | ■ Application Resp Code   |
| ■ Application Resp Flag   | ■ Application Resp Msg    |
| ■ Authorization Code      | ■ Authorization Resp Code |
| ■ Authorization Resp Flag | ■ Authorization Resp Msg  |
| ■ Authorization Response  | ■ Authorized Amount       |
| ■ Authorized Time         | ■ CV Result               |
| ■ Charge Resp Code        | ■ Charge Resp Flag        |
| ■ Charge Resp Msg         | ■ Charged Amount          |
| ■ Charged Time            | ■ Request Id              |
| ■ Row Id                  |                           |

- **CreditCard - Quote - AuthCharge (Siebel Input)**. Its integration component is Quote, with the following fields:

- Credit Card Transaction Amount
- Credit Card Transaction ID
- Credit Card Transaction Status
- Quote Number

This integration object is not used out of the box. You can use CreditCard - Quote - Authorization (Siebel Input) for this method.

### Refund

- **CreditCard - Refund (CyberSource Input)**. Its integration component is Invoice, with the following fields:

- |                   |                   |                          |
|-------------------|-------------------|--------------------------|
| ■ Amount          | ■ Bill Request Id | ■ Bill To City           |
| ■ Bill To Country | ■ Bill To State   | ■ Bill To Street Address |

- Bill To Zipcode
- Card Holder Name
- Card Start Month
- Duty Amount
- Merchant Descriptor Contact
- Phone Number
- Purchaser VAT Reg No
- Ship From Zipcode
- Summary Commodity Code
- VAT rate
- Card Expiration Month
- Card Issue Number
- Card Start Year
- Email Address
- Merchant Id
- Product Code
- Quantity
- Ship To Country
- Tax Amount
- Card Expiration Year
- Card Number
- Currency Code
- Merchant Descriptor
- National Tax
- Purchaser Code
- Row Id
- Ship To Zipcode
- Tax Indicator

■ **CreditCard - Refund (CyberSource Output).** Its integration component is Invoice, with the following fields:

- Application Resp Code
- Application Resp Msg
- Refund Auth Response
- Refund Resp Flag
- Refund Time
- Request Id
- Application Resp Flag
- Refund Amount
- Refund Resp Code
- Refund Resp Msg
- Refund Trans Ref Number
- Row Id

For more information on integration objects, see *Integration Platform Technologies: Siebel Enterprise Application Integration* and *Using Siebel Tools*.

## Data Maps

Data maps are used to map fields in one integration object to another integration object. The data maps used in credit card transactions are specified in the call to the business service.

You can view data maps by navigating to Integration Administration > Data Map Browser.

The data maps used in credit card transactions are shown in [Table 9](#).

Table 9. Data Maps Used in Credit Card Transactions

| Data Map                                  | Source Integration Object                          | Target Integration Object                         |
|---|--|---|
| <b>Authorization from Quote</b>           |  |   |
| CreditCardAuthorization_Quote2CyberSource | CreditCard - Quote - Authorization (Siebel Output) | CreditCard - Authorization (CyberSource Input)    |
| CreditCardAuthorization_CyberSource2Quote | CreditCard - Authorization (CyberSource Output)    | CreditCard - Quote - Authorization (Siebel Input) |

Table 9. Data Maps Used in Credit Card Transactions

| Data Map                                     | Source Integration Object                          | Target Integration Object                      |
|--|--|--|
| <b>Authorization from Payments</b>           |  |  |
| CreditCardAuthorization_Payments2CyberSource | CreditCard - Payments                              | CreditCard - Authorization (CyberSource Input) |
| CreditCardAuthorization_CyberSource2Payments | CreditCard - Authorization (CyberSource Output)    | CreditCard - Payments                          |
| <b>Charge from Payments</b>                  |  |  |
| CreditCardCharge_Payments2CyberSource        | CreditCard - Payments                              | CreditCard - Charge (CyberSource Input)        |
| CreditCardCharge_CyberSource2Payments        | CreditCard - Charge (CyberSource Output)           | CreditCard - Payments                          |
| <b>AuthCharge from Quote</b>                 |  |  |
| CreditCardAuthorization_Quote2CyberSource    | CreditCard - Quote - Authorization (Siebel Output) | CreditCard - Authorization (CyberSource Input) |
| CreditCardAuthCharge_CyberSource2Quote       | CreditCard - AuthCharge (CyberSource Output)       | CreditCard - Quote - AuthCharge (Siebel Input) |
| <b>AuthCharge from Payments</b>              |  |  |
| CreditCardAuthorization_Payments2CyberSource | CreditCard - Payments                              | CreditCard - Authorization (CyberSource Input) |
| CreditCardAuthCharge_CyberSource2Payments    | CreditCard - AuthCharge (CyberSource Output)       | CreditCard - Payments                          |
| <b>Refund from Payments</b>                  |  |  |
| CreditCardRefund_Payments2CyberSource        | CreditCard - Payments                              | CreditCard - Refund (CyberSource Input)        |
| CreditCardRefund_CyberSource2Payments        | CreditCard - Refund (CyberSource Output)           | CreditCard - Payments                          |

For more information on modifying integration objects and data maps, see *Integration Platform Technologies: Siebel Enterprise Application Integration* and *Using Siebel Tools*.

## CyberSource Adapter Service Business Service

The CyberSource Adapter Service business service does the following:

- Receives input from the Credit Card Transaction Service business service in XML format
- Processes the data and uses Siebel eScript to call the CyberSource API
- Gets the payment information from CyberSource using the CyberSource API

- Passes XML data back to the Credit Card Transaction Service business service

The CyberSource Adaptor Service consists of methods, user properties, and scripts written in Siebel eScript:

- Methods:
  - Authorization, with arguments inputXML and outputXML
  - Charge, with arguments inputXML and outputXML
  - Refund, with arguments inputXML and outputXML
  - Start, with argument outputXML
- Scripts:
  - **(declarations)**. This script defines the variables used in the business service scripts.
  - **CallCyberSource**. This script converts the XML to a property set, and then gets values from the property set to construct the message used by CyberSource. It calls CyberSource to get the result, uses the result to construct a property set, and then converts it to XML.
  - **FillOutIntObj**. This script constructs the property set from the CyberSource result.
  - **FillRequestObj**. This script constructs the CyberSource message from the property set.
  - **Initialize**. This script initializes the variables for the service. Variables to initialize are passed in by setting the values in the CreditCard - Configuration (CyberSource) integration object user properties.
  - **LogMessage**. This script writes to the log file the buffers that are passed to CyberSource and received from CyberSource, if the Debug flag is set.
  - **SendCmd**. This script calls CyberSource with a different method.
  - **Service\_PreInvokeMethod**. This is the starting point where the business service is called.
  - **SetResponseCode**. This script maps CyberSource error flags to Siebel response codes.
  - **ThrowError**. This script generates the error messages that are set up in the Credit Card Transaction Service business service user properties.

To examine the Siebel eScripts in the CyberSource Adapter Service business service, right-click the business service in Siebel Tools, and then choose Edit Server Scripts.

For more information on Siebel eScripts, see *Siebel Tools Online Help*.

## Address Verification with CyberSource

Address Verification will fail when CyberSource returns DAVSNO as the reply code. When DAVSNO is returned, an authorization code will still be returned to your Siebel Business Application. You can use the values for the `decline_avs_flags` business service user property to indicate which address verification results should result in DAVSNO being returned by CyberSource. From Siebel eSales, N is passed to CyberSource, indicating that when this result is received by CyberSource for the address verification check, DAVSNO will be returned to your Siebel Business Application. To indicate that other address verification responses should result in DAVSNO being returned to your Siebel Business Application, add the values to the `decline_avs_flags` business service user property. If you want to authorize and settle at the same time, you can indicate that address verification results should be ignored by setting the `ignore_avs` business service user property to `yes`.

## Customizing Cybersource Integration

You can customize Siebel eSales integration with CyberSource. Some examples are modifying scripts, passing more fields to CyberSource, and performing credit card processing on data in other business components.

This section contains the following topics:

- [“Modifying the Scripts in the CyberSource Adapter Service” on page 40](#)
- [“Passing Data from Other Fields in the Siebel Database” on page 41](#)
- [“Passing Data from Fields Not in the Siebel Database” on page 41](#)
- [“Calling the Credit Card Transaction Service from Other Places in Siebel eSales” on page 42](#)
- [“Calling the Credit Card Transaction Service from Other Business Components” on page 42](#)
- [“Using Other Third-Party Payment Applications” on page 42](#)

## Modifying the Scripts in the CyberSource Adapter Service

As the scripts in the CyberSource Adapter Service are written in Siebel eScript, they can be readily modified.

For more information on Siebel eScripts, see *Siebel Tools Online Help*. For more information on debugging scripts, see *Using Siebel Tools*.

### *To modify scripts in the CyberSource Adapter Service business service*

- 1 In Siebel Tools, lock the EAI CreditCard project.
- 2 Select the Business Service object.
- 3 Select the CyberSource Adapter Service, copy it, and then rename the copy as a backup.



- 4 Right-click CyberSource Adapter Service, and then choose Edit Server Scripts.  
The server script editor appears.
- 5 Edit the desired scripts, and then close the edit window.
- 6 Compile the .srf file, selecting the Locked projects radio button.

## Passing Data from Other Fields in the Siebel Database

You can add data fields to those passed to CyberSource.

For more information on modifying integration objects, see *Integration Platform Technologies: Siebel Enterprise Application Integration*. For more information on modifying scripts, see *Using Siebel Tools*.

### To add a data field

- 1 In Siebel Tools, add the field to the appropriate Cybersource input and output integration objects.
- 2 Modify the (declarations) script in the CyberSource Adapter Service business service to add the field to the correct output object, for example ICS2OBS["Authorization"].OutputFields.
- 3 In your Siebel Business Application, navigate to Integration Administration > Data Map Editor.
- 4 Update the data map so that data from the appropriate Siebel field will be passed to the CyberSource integration object.

## Passing Data from Fields Not in the Siebel Database

There are three ways to pass data from fields that do not exist in the Siebel database:

- Add the field to the database, add it to the necessary integration object, and then update the data map.
- Set an existing business service user property in the Credit Card Transaction Service business service, or create a new one, with a static value for all transactions. These values are read upon initialization of the business service.

Any data passed dynamically using the integration objects will overwrite the value in the business service user property for that particular transaction.

- Set a value for an integration object field by setting the value in the data map.

You cannot set a default value for an integration object user property because the Credit Card Transaction Service business service does not look for this value at initialization.

**NOTE:** The `auth_cv_result` reply field contains details of the card verification check. This data is not stored in the Siebel database but you can use it to store additional information or drive additional logic.

## Calling the Credit Card Transaction Service from Other Places in Siebel eSales

You can call a business service from any of the following:

- Button, as long as the applet that contains this button is based on the CSSBCBase class
- Workflow
- Script, either Siebel eScript or Siebel VB

### *To call the Credit Card Transaction Service business service*

- 1 Use the `GetService()` method to get an instance of the business service.
- 2 Use `Service_InvokeMethod()` to invoke a method of the business service.
- 3 Use `SetProperty()` to pass values for the business service method arguments that need to be set.  
See [“Business Service Methods” on page 29](#) for the business service method arguments of the Credit Card Transaction Service.

## Calling the Credit Card Transaction Service from Other Business Components

You can call the Credit Card Transaction Service business service from other business components besides Quote and Order.

### *To call the Credit Card Transaction Service from another business component*

- 1 Choose where to call the business service (button, workflow, or script).
- 2 In Siebel Tools, create integration objects for the business component to pass the correct data to the business service.
- 3 In your Siebel Business Application, navigate to Integration Administration > Data Map Editor.
- 4 Create integration object maps for the business component integration objects.

## Using Other Third-Party Payment Applications

Using other payment services requires modifying the CyberSource Adapter Service business service or writing a new one. Because the scripts in the CyberSource Adapter Service are in Siebel eScript, you can readily modify them.

For more information on Siebel eScripts, see *Siebel Tools Online Help*. For more information on business services and on debugging scripts, see *Using Siebel Tools*.

**NOTE:** After creating the new payment business service, you must create or modify integration objects and data maps for it.

*To create a new payment business service*

- 1** In Siebel Tools, lock the EAI CreditCard project.
- 2** Select the Business Service object.
- 3** Select the CyberSource Adapter Service and copy it.
- 4** Rename the copy, for example New Payment Adapter.
- 5** Right-click New Payment Adapter, and then choose Edit Server Scripts.  
The server script editor appears.
- 6** Modify the desired scripts, and then close the edit window.
- 7** Edit the business service user properties of the Credit Card Transaction Service Business Service to call the New Payment Adapter business service.
- 8** Compile the .srf file, selecting the Locked projects radio button.



# 5

## Integrating Siebel eSales with Third-Party Shipment Tracking Applications

This chapter tells you how to set up the ability for consumers to track the shipping progress of packages that you are sending to them. It contains the following sections:

- [“About Shipment Tracking Applications” on page 45](#)
- [“Process of Setting Up Shipment Tracking” on page 46](#)

### About Shipment Tracking Applications

The shipping carrier scans each package at various checkpoints on its journey, and stores information about the progress of the package in the carrier's shipment tracking database.

Customers can access the shipment tracking through Siebel eSales by clicking the Check Shipment button, which appears when there is a shipment. When the consumer clicks the button it shows the shipments that are associated with the order. If the shipment has a Shipment Identification Number, the Track Shipment button appears. When the consumer clicks this button, information in the shipment tracking database is delivered to Siebel eSales using a specific, prebuilt Application Services Interface (ASI) named External Shipment Tracking. Some configuration is required to set up this integration.

Integration also uses the published interfaces called Shipment Tracking Interface Request and Shipment Tracking Interface Response. For information about extending this integration object, and for more information about how to set up, use, and modify integrations based on ASIs, see *Integration Platform Technologies: Siebel Enterprise Application Integration*.

The following terms are used when referring to shipment tracking:

- **Shipment.** This is a part of an order that is shipped using the same carrier and shipment method. For example, an order can consist of one air-freight shipment and one ground shipment that have the same destination address. A shipment is made up of one or more packages. More than one order can be included in a single shipment, and a single order can require more than one shipment. In other words, there is a many-to-many relationship between orders and shipments.
- **Shipment Tracking Number.** Also known as a Shipment Identification number or Waybill number, this is a unique identifier used to track a shipment through the delivery system.
- **Package.** This is a container for the parts of a shipment. For example, a computer shipment may have a package for the monitor and a package for the computer's CPU case.
- **Package Tracking Number.** This is a unique identifier used to track a package through the delivery system.
- **Shipment Reference Number.** This is a unique identifier that is defined by the seller and used to track shipments. This number can be used in conjunction with a package, a set of packages within one or more shipments, or all packages within a shipment.

## Process of Setting Up Shipment Tracking

To set up the use of third-party shipment tracking system, you must perform the following tasks:

- 1 "Setting the Shipment Tracking-Package Tracking User Properties" on page 46.
- 2 "Verifying the External Shipment Tracking ASI User Properties" on page 47.
- 3 "Configuring Web Services for Shipment Tracking" on page 47.

To set up the use of third-party shipment tracking application, you can also perform the following optional tasks:

- "Extending Returned Data for Shipment Tracking" on page 48.
- "Changing the Package Sorting Order" on page 49.
- "Turning Off Package Sorting" on page 49.

## Setting the Shipment Tracking-Package Tracking User Properties

The virtual business component from which the shipment tracking application can be launched is associated with the Request Action method and its associated Request Option. These methods invoke the External Shipment Tracking ASI when clicking the Track Shipment button on the Shipment View (eSales) view. The Request Action is Track, and it should match the shipper's system command. The Request Option is Last Package or Last Activity, which should also match the shipper's system.

This task is a step in "Process of Setting Up Shipment Tracking" on page 46.

### *To set the Shipment Tracking - Package Tracking business component user properties*

- 1 In Siebel Tools, select the Shipment Tracking - Package Tracking business component.
- 2 Select Business Component User Prop.
- 3 Verify that the Names and Values of the User Properties match those in the following table.

| Business Component                 | Name           | Default Value |
|------------------------------------|----------------|---------------|
| Shipment Tracking-Package Tracking | Request Action | Track         |
|                                    | Request Option | Last Activity |

**NOTE:** If you want to support integration with more than one carrier, the user property values required by the carriers will probably be different. In this case, you must employ a middleware adapter that will change the values passed to each carrier.

## Verifying the External Shipment Tracking ASI User Properties

You must verify the External Shipment Tracking ASI user properties.

This task is a step in [“Process of Setting Up Shipment Tracking”](#) on page 46.

### *To verify the External Shipment Tracking ASI user properties*

- 1 In Siebel Tools, select the External Shipment Tracking business service.
- 2 Select Business Service User Prop.
- 3 Verify that the Names and Values of the User Properties match those in the following table.

| Name                         | Value                      |
|------------------------------|----------------------------|
| siebel_port_name             | Default                    |
| siebel_web_service_name      | External Shipment Tracking |
| siebel_web_service_namespace | http://siebel.com/asi/     |

## Configuring Web Services for Shipment Tracking

Web services are used to direct the inbound and outbound messages to and from:

- The third-party shipment tracking application.
- A demonstration script that can be used to test your integration setup.
- A middleware component.

You must make sure that the Web services are correctly configured.

This task is a step in [“Process of Setting Up Shipment Tracking”](#) on page 46.

### *To configure the Web services*

- 1 Navigate to the Administration - Web Services > Outbound Web Services.
- 2 Select the row with External Shipment Tracking in the Name field.
- 3 Verify that the fields in the Outbound Web Services list match those in the following table.

| Namespace              | Name                       | Status |
|------------------------|----------------------------|--------|
| http://siebel.com/asi/ | External Shipment Tracking | Active |

- 4 In the Service Ports list, click New, and enter the following values in the fields of the new record.

| Field                | Comments   |
|----------------------|--|
| Name                 | Enter a name for the port.   |
| Service Display Name | Enter External Shipment Tracking.  |
| Transport            | Select from the drop-down list.<br><br>If you want to set up testing, for example when you are customizing applets or extending the ASI, select Local Business Service.  |
| Address              | When you select a transport, the field is populated with a template address. Edit the address based on where you are sending the data.<br><br>If you want to set up testing, enter Shipment Tracking ASI Test. |
| Binding              | If a SOAP header is expected by the receiving application, select SOAP_RPC.<br><br>If you are using Shipment Tracking ASI Test, select PropertySet.  |

For more information about configuring Web Services and setting up demonstration scripts, see *Integration Platform Technologies: Siebel Enterprise Application Integration*.

## Extending Returned Data for Shipment Tracking

You can extend the returned data by adding a field to the response integration object and displaying it in the user interface.

This task is a step in [“Process of Setting Up Shipment Tracking”](#) on page 46.

### *To extend the returned data*

- 1 Add the field to the Shipment Tracking Interface Response integration object.
- 2 Add the field to the Shipment Tracking - Package Tracking business component, making sure that the name of the field matches that of the integration object.
- 3 Add the control to the Shipment Tracking - Package Tracking (eSales) applet, which maps to the added field in the business component.

For more information about extending the data that is received from the shipment tracking application, see *Integration Platform Technologies: Siebel Enterprise Application Integration*.



## Changing the Package Sorting Order

Packages are displayed as a list of records in the Shipment Tracking - Package Tracking (eSales) applet. The list is sorted in the order shown in [Table 10](#).

Table 10. Package Sorting Order

| Sort Level | Value Sorted            | Sort Order |
|------------|-------------------------|------------|
| First      | Package Tracking Number | Ascending  |
| Second     | Date                    | Descending |
| Third      | Time                    | Descending |

You can change the order in which packages are listed.

This task is a step in [“Process of Setting Up Shipment Tracking” on page 46](#).

### *To change the package sorting order*

- 1 In Siebel Tools, select the Shipment Tracking (eSales) business service.
- 2 Select Business Service User Prop.  
The displayed list of user properties includes nine sort fields.
- 3 Change the values of the fields to change the sorting order for the three levels of sort.

## Turning Off Package Sorting

You may want to turn off package sorting to improve performance if:

- The order in which information is returned does not matter to you.
- You usually only have one package in a shipment.
- Sorting is performed in middleware.
- Sorting is performed by the carrier’s shipment tracking application.

This task is a step in [“Process of Setting Up Shipment Tracking” on page 46](#).

### *To turn off package sorting*

- 1 In Siebel Tools, select the Shipment Tracking-Package Tracking business component.
- 2 Select Business Component User Prop.
- 3 Select the Sort Results user property.
- 4 Change the value from Y to N.



# 6

## Administering Siebel eSales

This chapter describes administrative tasks used to maintain Siebel eSales. These tasks include setting up quote expiration dates, setting up customer accounts, and modifying credit card charges.

This chapter contains the following sections:

- ["Setting Quote Expiration Dates" on page 51](#)
- ["Managing Anonymous Shopping Carts" on page 51](#)
- ["About Setting Up Customer Accounts" on page 52](#)
- ["Modifying Credit Card Charges" on page 54](#)

### Setting Quote Expiration Dates

Quote expiration dates are set in the List of Values view under the Application Administration screen.

**NOTE:** You must clear the LOV cache for the new expiration date to take effect.

#### *To set quote expiration dates*

- 1 In your Siebel employee application, navigate to Administration - Data > List of Values.
- 2 Query for QUOTE\_EXPIRATION\_PERIOD in the Type field.
- 3 Set the Display Value field to the number of days you want quotes to be effective.
- 4 Click Menu and choose Clear Cache.

### Managing Anonymous Shopping Carts

Customers are treated as anonymous users until they register with your Siebel eSales Web site. Registered users returning to your Web site, and who have not yet logged in, are recognized as anonymous users rather than registered users if:

- They have deleted previously stored cookies.
- They elected for the Web site not to remember their user IDs, in which case they lack the cookie that enables implicit login.

When an anonymous user places an item in the Shopping Cart, an anonymous quote is created and a cookie is placed on the customer's computer. The cookie identifies the quote. If the Internet connection is lost, or if the customer leaves your Web site before checking out, the cookie is used to load the customer's quote when the customer returns.

If the customer does not return to the Web site for a specified number of days, the anonymous quote is deleted.

If the customer logs in later, and before the quote is deleted, the persistent anonymous quote is converted to a persistent quote, and the cookie is cleared. The cookie is replaced by another one that will enable implicit login when the customer returns to your site.

You can set both the frequency with which anonymous quotes are deleted and the age of the quotes to be deleted. The deletions are then performed automatically by a repeating server request. By default, anonymous shopping carts 30 days or older are deleted weekly. For information about setting up repeating server requests, see *Siebel Business Process Designer Administration Guide*.

### Setting the Age for Anonymous Quotes Deletion

To set the age for anonymous quotes deletion, do the following in Siebel Tools:

- Select the eSales - Delete Old Anonymous Quotes workflow.
- Locate the workflow's Days Old process property and edit its Default String field.
- Make any other changes to the workflow that are necessary for your implementation.

For detailed information on modifying workflows, see the chapter on building workflow processes in *Siebel Business Process Designer Administration Guide*.

## About Setting Up Customer Accounts

The tasks involved in setting up customer accounts are as follows:

- Creating accounts.

For more information, see the chapter on using accounts in Siebel Sales in *Siebel Applications Administration Guide*.

- Associating users with accounts.

You assign an account's users by filling in the account's Contacts field. For more information, see the chapter on using accounts in Siebel Sales in *Siebel Applications Administration Guide*.

**NOTE:** When you create a Siebel eSales user, the Contact Team for the user belongs to your organization. The Contact Team should be set to the correct position for the Organization to which the user belongs, so the user gets the correct price list.

- Associating price lists with accounts.

You assign a price list to an account by filling in the account's Price List field. For more information, see the chapter on using accounts in Siebel Sales in *Siebel Applications Administration Guide*.

- Allowing accounts to use purchase orders.

You allow an account to use purchase orders by selecting the account's PO Approved check box. For more information, see ["About Allowing Accounts and Contacts to Use Purchase Orders" on page 53](#) and the chapter on using accounts in Siebel Sales in *Siebel Applications Administration Guide*.

- Allowing contacts to use purchase orders.

For more information, see [“About Allowing Accounts and Contacts to Use Purchase Orders” on page 53](#) and [“Allowing Contacts to Use Purchase Orders” on page 54](#).

- Setting purchase order auto-approval limits for accounts.

The PO auto-approval limit is the maximum amount that a corporate customer can order using a purchase order without approval from the purchasing manager in the purchaser’s company. For more information, see the chapter on using accounts in Siebel Sales in *Siebel Applications Administration Guide*.

- Setting credit auto-approval limits for accounts.

The credit auto-approval limit is the maximum amount that a corporate customer can order using a purchase order without requiring an automatic credit check using the External Credit Check ASI. For example, if you set the limit to \$1000, any order greater than \$1000 automatically receives a credit check. For more information, see the chapter on preparing for integration in *Siebel Order Management Guide*.

- Allowing accounts to skip credit checks.

You can skip automatic credit checking for a particular account. Automatic credit checking uses the External Credit Check ASI for customers paying with a purchase order. For more information, see the chapter on preparing for integration in *Siebel Order Management Guide*.

## About Allowing Accounts and Contacts to Use Purchase Orders

Corporate customers can use purchase orders to buy from your Siebel eSales Web site. Web Corporate Users who want to make purchases may not have the authority to assign a purchase order number and approve the purchase for their company. Siebel eSales allows Web Corporate Users to complete the checkout process without providing a purchase order number. When this happens, the PO Status is set to Pending and the Order Status is set to Awaiting Approval. This happens even if the order amount is less than the Auto-Approval Limit set on the account. The order is then routed through the order approval process.

The Web Delegated Customer Administrator in the buying organization can set purchase order approval for particular users within the buying organization. The Web Purchasing Manager in the buying organization can review orders that are awaiting approval through the My Company’s Orders screen. The Web Purchasing Manager can then provide a purchase order number and change the Order Status to Approved.

To allow Web Corporate Users to buy with purchase orders, the PO Approved flag must be set for both of the following:

- The account
- All contacts associated with the account who should be able to buy with a purchase order

There is no inheritance from an account to its contacts.

**NOTE:** The Web Purchasing Manager must also be assigned a user type of Web Purchasing Manager and the Web Corporate User responsibility. For more information, see [“Setting Up Default Responsibilities and User Types for eSales”](#) on page 16.

## Allowing Contacts to Use Purchase Orders

Use this procedure to allow a contact to use a purchase order.

### *To allow a contact to use purchase orders*

- 1 In your Siebel employee application, navigate to Administration - User > Users.
- 2 In the Users list, select a user whom you want to allow to use purchase orders.
- 3 In the form, select the PO Approved check box.

## Modifying Credit Card Charges

Call center or sales professionals can modify credit card charges in your Siebel Business Application. The tasks associated with modifying credit card charges are as follows:

- **Authorize.** This task allocates to the merchant a specific amount of the credit available to the cardholder, typically for one week. No funds are transferred.

The merchant authorizes a credit card to make sure of the following:

- The customer has the funds to pay for the purchase.
- These funds are reserved until the merchant is able to ship the product. Credit card companies typically state that a merchant should not settle until the product has been shipped.

Siebel eSales does automatic authorization as part of the eSales - Authorize Order Process, when the customer clicks Confirm Order on the Order Summary page. Authorization can also be done manually. Your employees, such as call center agents, use the Payment form to authorize charges.

For more information, see the chapter on creating a quote or order in *Siebel Order Management Guide*.

- **Settle.** This task includes the authorization code and requests that funds be transferred from the cardholder's account to the merchant's bank account.

You settle charges after an order has been authorized. You will need to settle each order or develop a batch routine.

For more information, see the chapter on creating a quote or order in *Siebel Order Management Guide*.

- **Authorize and Settle.** This task makes sure that the funds are available, reserves them, and requests that they be transferred in one call. It does this in the case of immediate order fulfillment (for example, digital downloads).  
For more information, see the chapter on creating a quote or order in *Siebel Order Management Guide*.
- **Refund.** This task takes place after a settlement when the merchant needs to transfer money back to the cardholder, as in the case of a return.  
For more information, see the chapter on creating a quote or order in *Siebel Order Management Guide*.
- **Reversal.** This task takes place as a result of allowing a transaction that was not settled to expire. See [“Reversing Charges” on page 55](#).
- **Identifying Errors.** See [“Querying For Transactions With Error Messages” on page 55](#).

## Reversing Charges

Reversing charges releases the amount of money reserved by an authorization. This transaction is important because authorizations reserve part of a cardholder's credit and therefore reduce the total amount of credit available.

**NOTE:** You cannot reverse charges using the CyberSource Internet Commerce Suite. Authorizations are allowed to expire after not being settled for a specified period of time, usually one week.

## Querying For Transactions With Error Messages

You can query for credit card authorizations that returned Siebel Response Messages without stopping the ordering process.

Siebel Response Messages that do not stop the ordering process are set in the Create Order on Web business service user property for the Credit Card Transaction Service business service. These messages are listed in [Table 11](#).

Table 11. Siebel Response Messages

| Error Number | Message  |
|--------------|--|
| 1            | Transaction was successful.  |
| 18           | You are not authorized to perform this transaction type.                                   |
| 19           | Error happened in adapter business service. Please contact your application administrator. |
| 20           | Error happened in CyberSource. Please contact your application administrator.              |
| 21           | The connection to CyberSource has timeout. Please try the operation later.                 |

For more information, see [Chapter 4, "Integrating Siebel eSales with Third-Party Payment Applications."](#)

***To query for Siebel Response Messages***

- 1** In your Siebel employee application, navigate to Sales Orders > List.
- 2** Select an order and click its Order # hyperlink.
- 3** Click the Payments view tab.

Returned Siebel Response messages are displayed in the Transaction Message column of the Payment Lines list.



# 7

## Siebel eSales Shopping Cart

This chapter describes the Shopping Cart and introduces the Shopping Service business service. It contains the following sections:

- [“About eSales Shopping Cart” on page 57](#)
- [“Quote Types in Siebel eSales” on page 57](#)
- [“Shopping Service Business Service” on page 58](#)
- [“Run-Time Events in Siebel eSales” on page 60](#)
- [“Object Manager Support for Shopping Cart” on page 62](#)

### About eSales Shopping Cart

The Shopping Cart stores items selected by customers. When a Siebel eSales customer selects an item for the first time during a session by clicking Add to Cart, a quote record is created. The CurrentQuoteId user profile attribute is set to the ID of this quote.

A customer can have only one Shopping Cart in a session. If the customer changes the Shopping Cart to another quote, for example in the Edit Quote workflow, CurrentQuoteId is set to the ID of the new quote record.

### Quote Types in Siebel eSales

Several quote types support anonymous users and persistent Shopping Carts. The quote types used in Siebel eSales are as follows:

- **Anonymous Quote.** This type allows anonymous users to add items to the Shopping Cart. When an anonymous user logs in or registers, the quote type of the current quote is checked and converted to Persistent Cart. This preserves the Shopping Cart after login.

You can use workflows to delete Anonymous Quote records if they last for more than a specified period of time, such as 48 hours. This would apply if the anonymous user chose not to log in.

- **Persistent Cart.** This is the persistent Shopping Cart. Each user can have only one quote record of type Persistent Cart.

The persistent Shopping Cart works as follows:

- When a user logs in, the run-time event WebLogin triggers the action set eSales Web Login Action Set, which invokes the Shopping Service method SetShoppingCart.
- SetShoppingCart finds the quote with type Persistent Cart associated with that user and sets CurrentQuoteId to the ID of this quote.

- **Quotation.** When a user clicks Confirm Order on the Order Summary page, the quote type changes from Persistent Cart to Quotation and the status changes to Order Placed.

## Shopping Service Business Service

The Shopping Cart is implemented through the Shopping Service business service. The Shopping Service business service is based on the class CSSShoppingService, a subclass of CSService. The user interface invokes some of its methods, while others are invoked by workflows, or by applications such as Siebel eEvents Management.

This section contains the following topics:

- [“Customizing the Shopping Service” on page 58](#)
- [“Shopping Service Business Service Methods” on page 58](#)

### Customizing the Shopping Service

The Shopping Service business service is written in C++ and cannot be modified. If you want to change any of the methods in the Shopping Service, you need to write a new business service in Siebel eScript. However, these methods are modular and they can be used in new workflows that are designed when the application is customized.

The following sections provide more information on the Shopping Service and other processes used in Siebel Business Applications to support the Shopping Cart.

For more information on writing scripts, see *Siebel Tools Online Help*.

### Shopping Service Business Service Methods

The Shopping Service has many business service methods. Those used by the Shopping Cart in Siebel eSales are described in [Table 12](#).

For detailed descriptions of these methods, see [Appendix A, “Shopping Service Methods Reference.”](#)

Table 12. Shopping Service Business Service Methods

| Method   | Description   | Used In Workflow                 |
|--|---|----------------------------------|
| <a href="#">“AddItem Method” on page 112</a>     | Adds an item to the Shopping Cart or Favorites.                                     |                                  |
| <a href="#">“CreateOrder Method” on page 138</a> | Converts a quote to an order. Also confirms training class and event registrations. | eSales - Authorize Order Process |

Table 12. Shopping Service Business Service Methods

| Method   | Description   | Used In Workflow   |
|--|---|--|
| <a href="#">“DiscardCart Method” on page 133</a>           | Deletes the current Shopping Cart and sets CurrentQuoteId to empty, but only when the Quote Type equals Persistent Cart.  | eSales - Edit Quote Process<br>eSales - Reorder Process                                  |
| <a href="#">“GetShoppingCartStatus Method” on page 142</a> | Gets the status of the current Shopping Cart, which can be one of the following: <ul style="list-style-type: none"> <li>■ Empty. CurrentQuoteId is empty or Quote Type equals Persistent Cart.</li> <li>■ Not Empty.</li> <li>■ No Save.</li> </ul> Also returns information such as Quote Item and Item Count. | eSales - Edit Quote Process<br>eSales - Modify Order Process<br>eSales - Reorder Process |
| <a href="#">“OrderQuote Method” on page 135</a>            | Creates an order based on the quote.  | eSales - Checkout Process<br>eSales - Order Quote Process                                |
| <a href="#">“Reorder Method” on page 140</a>               | Creates a quote based on an existing order.   |  |
| <a href="#">“RepriceQuote Method” on page 144</a>          | Reprices quotes, for example when a customer orders a saved quote that has expired.   |  |
| <a href="#">“SaveCartAsQuote Method” on page 131</a>       | Saves the current Shopping Cart as a quote: <ul style="list-style-type: none"> <li>■ Finds the quote record whose Id equals CurrentQuoteId.</li> <li>■ Saves it as a quote, that is, sets the quote status to Quotation.</li> <li>■ Sets CurrentQuoteId to empty (empties the Shopping Cart).</li> </ul>        | eSales - Edit Quote Process<br>eSales - Reorder Process<br>eSales - Save Cart Process    |
| <a href="#">“SaveCartAsTemplate Method” on page 129</a>    | Saves the current Shopping Cart as a template, but keeps the Shopping Cart.<br><br>Ordering from a template copies the template items to the Shopping Cart.   | eSales - Save Cart Process   |
| <a href="#">“SelectPrimary Method” on page 145</a>         | Uses the contact Primary ID field to copy the primary personal address, primary personal account address, and primary payment profile from the pick field to the current quote (Shopping Cart).   |  |

Table 12. Shopping Service Business Service Methods

| Method  | Description  | Used In Workflow   |
|---|--|--|
| <a href="#">“SetApplicationName Method” on page 124</a> | <p>Sets the ApplicationName dynamic user profile attribute. Triggered by the application’s WebLogin event.</p> <p>Used in run-time events to support the different Shopping Cart behavior between Siebel eSales and other Siebel Business Applications.</p>  |  |
| <a href="#">“SetPendingRequest Method” on page 148</a>  | <p>Allows workflows to continue after anonymous users are sent to the Login view to register.</p>  | <p>eSales - Checkout Process</p> <p>eSales - Save Cart Process</p> |
| <a href="#">“SetShoppingCart Method” on page 122</a>    | <p>Supports the persistent Shopping Cart and allows anonymous users to add items to the Shopping Cart. Triggered by the application’s WebSessionStart event.</p> <p>When a session begins, the method does the following:</p> <ul style="list-style-type: none"> <li>■ If the user is anonymous, does nothing.</li> <li>■ If CurrentQuoteId is not empty (that is, the user already has a Shopping Cart), checks if the Shopping Cart quote type is Anonymous Quote.</li> </ul> <p>If there is an Anonymous Quote, converts it to Persistent Cart and sets Contact Id to the login Id of the user.</p> <ul style="list-style-type: none"> <li>■ If CurrentQuoteId is empty, tries to find an existing quote of type Persistent Cart (user’s Shopping Cart from the previous session) and set CurrentQuoteId to that quote ID.</li> </ul> <p>If there is no such quote, the user did not leave a Shopping Cart (for example, saved it as a quote or checked out).</p> |  |

## Run-Time Events in Siebel eSales

Tracking run-time events allows Siebel Business Applications to respond in real time to user actions. Business managers can configure the actions in response to any event.

When a run-time event occurs that is associated with one or more action sets, the personalization engine performs the specified actions. These actions modify content and user profile attributes, with the potential of triggering personalization rules.

Run-time events and action sets are created and managed in the Run-time Events Administration screen. You do not need to reconfigure or recompile business objects, because the interaction occurs between Siebel Personalization and the user interface.

Siebel eSales uses three action sets to configure Shopping Cart behavior:

- **eSales WebSessionStart Action Set.** This is triggered by the WebSessionStart run-time event. This action set contains one action:

**SetApplicationName.** This action sets the ApplicationName dynamic user profile attribute, which supports the different behavior of the Shopping Cart among the various Siebel Business Applications.

The following applications use this action set:

- Siebel Call Center
- Siebel Partner Relationship Management
- Siebel eCustomer
- Siebel eEvents Management
- Siebel eSales
- Siebel Self Service
- Siebel Marketing

- **eSales WebLogin Action Set.** This is triggered by the run-time event WebLogin. It contains the following actions:

- **SetPriceList.** This action sets the price list for the session when a user logs in. It invokes the Pricing Manager business service method SetPriceList, which sets the price list according to the following:

- If the user belongs to an account, the price list associated with that account is used.
- If the user does not belong to an account or belongs to an account that is not associated with a price list, the price list whose row ID is in the PRICER\_MASTER\_PRICE\_LIST\_ID list of values is used. For more information, see [“Setting Up Price Lists for Siebel eSales” on page 18.](#)

- **SetShoppingCart.** This action determines, when a user logs in, whether that user has a Shopping Cart from a previous session or from browsing as an anonymous user. It invokes the Shopping Service business service method SetShoppingCart, which checks the user profile attribute CurrentQuoteId for the following:

- If CurrentQuoteId is null, SetShoppingCart finds the quote record of type Persistent Cart, if any, and sets CurrentQuoteId to this record’s ID.

- If CurrentQuoteId is not null, and the user has a quote record of type Anonymous Quote, SetShoppingCart calls the ConvertAnonymousQuote method to change the quote type to Persistent Cart.

Three applications use the eSales WebLogin Action Set:

- Siebel Partner Relationship Management
- Siebel eCustomer
- Siebel eSales
- **Workflow\_Checkout.** This is conditionally triggered by the run-time event WebLogin. It is triggered only if an anonymous user tries to check out and is redirected for login. After login, the event is triggered to complete the checkout process. It contains the following actions:
  - **Workflow.** This action calls the SS Start Checkout Process workflow (see [Figure 8 on page 76](#)) to complete the checkout process.

Four applications use the Workflow\_Checkout Action Set:

- Siebel Consumer Marketing
- Siebel Consumer Sales
- Siebel Consumer Web
- Siebel CX Partner Portal

For more information on run-time events, see *Siebel Personalization Administration Guide*.

## Object Manager Support for Shopping Cart

Siebel Business Applications use object manager parameters to support different Shopping Cart requirements. The default values of object manager parameters are defined in Siebel Tools but these can be overridden by individual applications. The method of overriding the parameter values is different for the Siebel Web client and Siebel mobile client.

This section contains the following topics:

- [“Viewing Default Object Manager Parameters” on page 62](#)
- [“Modifying Object Manager Parameters in the Web Client” on page 63](#)
- [“Modifying Object Manager Parameters in the Mobile Web Client” on page 64](#)

### Viewing Default Object Manager Parameters

You can view the default object manager parameters in Siebel Tools.

**To view the default object manager parameters**

- 1 In Siebel Tools, if the Server Component Type object is not shown in the Object Explorer:
  - a Select View > Options.
  - b In the Development Tools Options dialog box, click the Object Explorer tab.
  - c Select the Server Component Type object in the left pane and move it to the right pane.
- 2 In the Object Explorer, select Server Component Type.
- 3 Query for InfraShoppingService.
- 4 In the Object Explorer, expand Server Component Type and select Server Component Parameter.  
The default server component parameters are displayed.

## Modifying Object Manager Parameters in the Web Client

You can see or change the definitions of overrides in the Web client.

**To see or change the definitions of overrides**

- 1 In your Siebel employee application, navigate to Administration - Server Configuration > Servers.
- 2 Make sure the Components tab is selected.
- 3 Select the appropriate component.  
For example, if you want to see the eSales component, select eSales Object Manager.
- 4 Click the Component Parameters view tab.
- 5 Examine the component parameters that are listed in Siebel Tools.

Siebel Sales and Siebel Call Center use the default parameters. Siebel eSales and Siebel Partner Relationship Management parameter override values are shown in [Table 13](#). In all cases, Condition is =.

Table 13. Parameters for Siebel Partner Relationship Management

| Parameter              | eSales Override Value                   | PRM Override Value          |
|------------------------|---|-----------------------------|
| ProductDetailView      | Product Detail - Features View (eSales) | FALSE                       |
| ShoppingCartView       | SS Quote Shopping Cart View             | SS Quote Shopping Cart View |
| AnonymousQuote         | TRUE                                    | TRUE                        |
| PersistentShoppingCart | TRUE                                    | TRUE                        |

## Modifying Object Manager Parameters in the Mobile Web Client

For users of the Siebel Mobile Web Client, the application configuration files, shown in [Table 14](#), are used to change object manager parameters.

Table 14. Application Configuration Files

| Application                            | Configuration File |
|--|--------------------|
| Siebel eSales                          | esales.cfg         |
| Siebel Partner Relationship Management | scw.cfg            |
| Siebel Call Center                     | uagent.cfg         |
| Siebel Sales                           | siebel.cfg         |



# 8

## Siebel eSales Workflow Reference

This chapter describes the seed-data workflows provided by Siebel eSales. You can use these workflows as starting points for workflow customization, based on your company's business needs.

This chapter contains the following sections:

- ["About Siebel eSales Workflows" on page 65](#)
- ["Activating Siebel eSales Workflows" on page 66](#)
- ["Siebel eSales Quote and Template Workflows" on page 67](#)
- ["Siebel eSales Checkout Workflows" on page 74](#)
- ["Siebel eSales Notification Workflows" on page 96](#)
- ["Siebel eSales Purchase Approval Workflows" on page 97](#)
- ["Siebel eSales Order Modification Workflows" on page 100](#)
- ["Siebel eSales Error Message and View Display Workflows" on page 103](#)
- ["Auction Workflow" on page 108](#)

### About Siebel eSales Workflows

Siebel eSales uses workflows to control order management processes. You can modify these workflows using Siebel Business Process Designer, which is in Siebel Tools.

For more information about modifying workflows, see *Siebel Business Process Designer Administration Guide*.

To modify the error messages that eSales displays, use a subprocess call to the eSales - Display Error Process (see [Figure 32 on page 104](#)).

The following features are among those implemented by workflow processes:

- **Rapid Checkout.** This feature automatically associates primary address and credit card information with the order during the checkout process. If the user does not have primary information stored within the application, the process directs that user to submit such information.
- **Purchase Approval Process.** This feature applies an approval process in a business-to-business environment under the following conditions:
  - The account is approved to pay by purchase order and a purchase order is used as the payment method.
  - The amount of the order is greater than the auto-approval limit.
  - A user of type Web Corporate User placed the order.

- **Email Notifications.** This feature sends an email notification to a user when that user's order is accepted, awaiting approval, rejected, or shipped.

## Activating Siebel eSales Workflows

Before you can use Siebel eSales, you must activate its workflows. For information about activating workflows, see *Siebel Business Process Designer Administration Guide*.

Siebel eSales uses the following workflows:

- Quotes and Templates
  - eSales - Edit Quote Process on page 68
  - eSales - Order Quote Process on page 70
  - eSales - Reprice Process on page 71
  - eSales - Save Cart Process on page 72
  - eSales - Add To Template Process on page 73
  - eSales - Delete Old Anonymous Quotes on page 74
- Checkout
  - SS Start Checkout Process on page 76
  - Credit Check - Quotes (eSales) on page 77
  - eSales - Enter Shipping Details Process on page 78
  - SS Save Shipping Address Details Process on page 80
  - eSales - Enter Payment Details Process on page 82
  - SS Save Payment Details Process on page 84
  - eSales - Complete Checkout Process on page 86
  - eSales - Authorize Order Process on page 87
  - SS Authorize Order Process on page 89
  - eSales - Save Credit Card on page 90
  - SS Set Shipping Checkout Mode Process on page 91
  - eSales - Save Shipping Address on page 92
  - eSales - Save Account Shipping Address on page 92
  - eSales - Set Primary Account Address on page 93
  - eSales - Setting PO Status on page 94
  - eSales - Current User Type on page 95
  - SS Change Checkout Contact Detail Process on page 95
  - SS Cancel Checkout Process on page 96

- Purchase approval
  - [eSales - Approve Order Process on page 98](#)
  - [eSales - Reject Order Process on page 99](#)
- Order modification
  - [eSales - Cancel Order Process on page 100](#)
  - [eSales - Modify Order Process on page 101](#)
  - [eSales - Reorder Process on page 102](#)
- Error messages and views
  - [eSales - Display Error Process on page 103](#)
  - [eSales - Login View on page 105](#)
  - [eSales - Order Confirmation View on page 106](#)
  - [SS Order Confirmation View Process on page 107](#)
  - [eSales - Shopping Cart View on page 108](#)
- Auctions
  - [eSales - Silent Authorize Order Process on page 109](#) (included for backward compatibility only)

In addition to the workflows covered in this chapter, Siebel eSales uses [“Shipping Calculation Process Workflow” on page 21](#).

#### ■ Notifications

If your company wants to use Siebel’s email notification functionality in addition to standard eSales functionality, then you must also activate the following workflows:

- [Send Order Accepted Email \(eSales\)](#)
- [Send Order Awaiting Approval Email \(eSales\)](#)
- [Send Order Failed Email \(eSales\)](#)
- [Send Order Rejected Email \(eSales\)](#)
- [Send Order Shipped Email \(eSales\)](#)

## Siebel eSales Quote and Template Workflows

A quote is an offer to sell a set of items at particular prices for a defined period of time. A template is a group of items that are frequently purchased together.

The Shopping Cart can be saved as a quote or a template. You can edit saved quotes at any time. If a quote has not expired, it can also be ordered directly.

This section describes the following workflows:

- [eSales - Edit Quote Process on page 68](#)

- eSales - Order Quote Process on page 70
- eSales - Reprice Process on page 71
- eSales - Save Cart Process on page 72
- eSales - Add To Template Process on page 73
- eSales - Delete Old Anonymous Quotes on page 74

## eSales - Edit Quote Process

The workflow for editing a saved quote is shown in Figure 2.

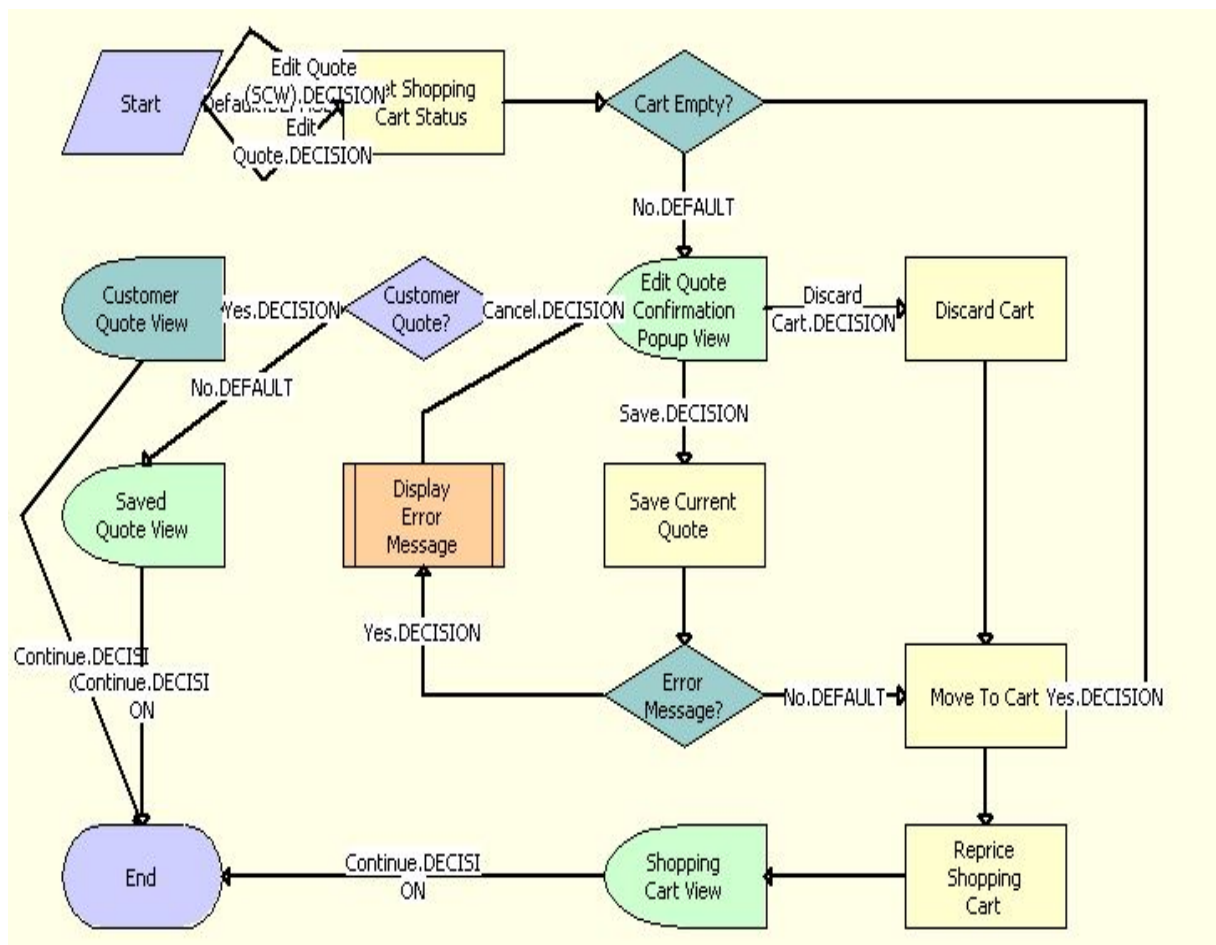


Figure 2. eSales - Edit Quote Process

The eSales - Edit Quote Process is called when a user clicks Edit Quote in the Saved Quote Detail view. The following events happen:

- The process determines whether the Shopping Cart is empty.
- If it is empty, the quote is moved to the Shopping Cart.
- If it is not empty, the Edit Quote Confirmation pop-up view appears.

The user has three choices:

- **Cancel.** The process determines whether the quote is for an existing customer. If it is, the Customer Quote Detail View appears and the process ends. If it is not, the Saved Quote Detail View (eSales) view appears and the process ends.
- **Discard Cart.** The Shopping Cart is emptied and the quote to be edited is moved to the Shopping Cart.
- **Save.** If there is an error, the eSales - Display Error Process (see [Figure 32 on page 104](#)) is called, and then the user is sent back to the Edit Quote Confirmation pop-up view. For example, this would happen if the user tried to save a quote with a name that was already being used.

If there is no error, the Shopping Cart is saved as a quote and the quote to be edited is moved to the Shopping Cart (see ["eSales - Save Cart Process" on page 72](#)).

- The Shopping Cart is repriced and displayed.



- ❑ **Discard Cart.** The current contents of the Shopping Cart are discarded. The quote is moved into the Shopping Cart and repriced, and then the user is sent to the Shopping Cart view.
- ❑ **Save.** The current contents of the Shopping Cart are saved as a quote. If there is an error, the eSales - Display Error Process (see [Figure 32 on page 104](#)) is called, and then the user is sent back to the Order Quote Confirmation view. For example, this would happen if the user tried to save a quote with a name that was already being used.

If there is no error, the quote is moved into the Shopping Cart and repriced, and then the user is sent to the Shopping Cart view.

## eSales - Reprice Process

The workflow for repricing a quote is shown in [Figure 4](#).

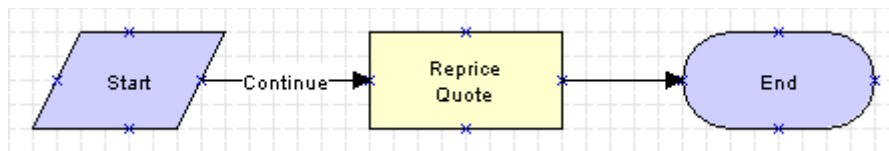


Figure 4. eSales - Reprice Process

The items in the quote are repriced based upon the current price in the appropriate price list.

The eSales - Reprice Process is called by the eSales - Modify Order Process (see [Figure 30 on page 101](#)). Repricing a quote is its own process because the eSales - Reprice Process is based on the Catalog business object, while the eSales - Modify Order Process is based on the Order Entry (Sales) business object.

## eSales - Save Cart Process

The workflow for saving a Shopping Cart is shown in Figure 5.

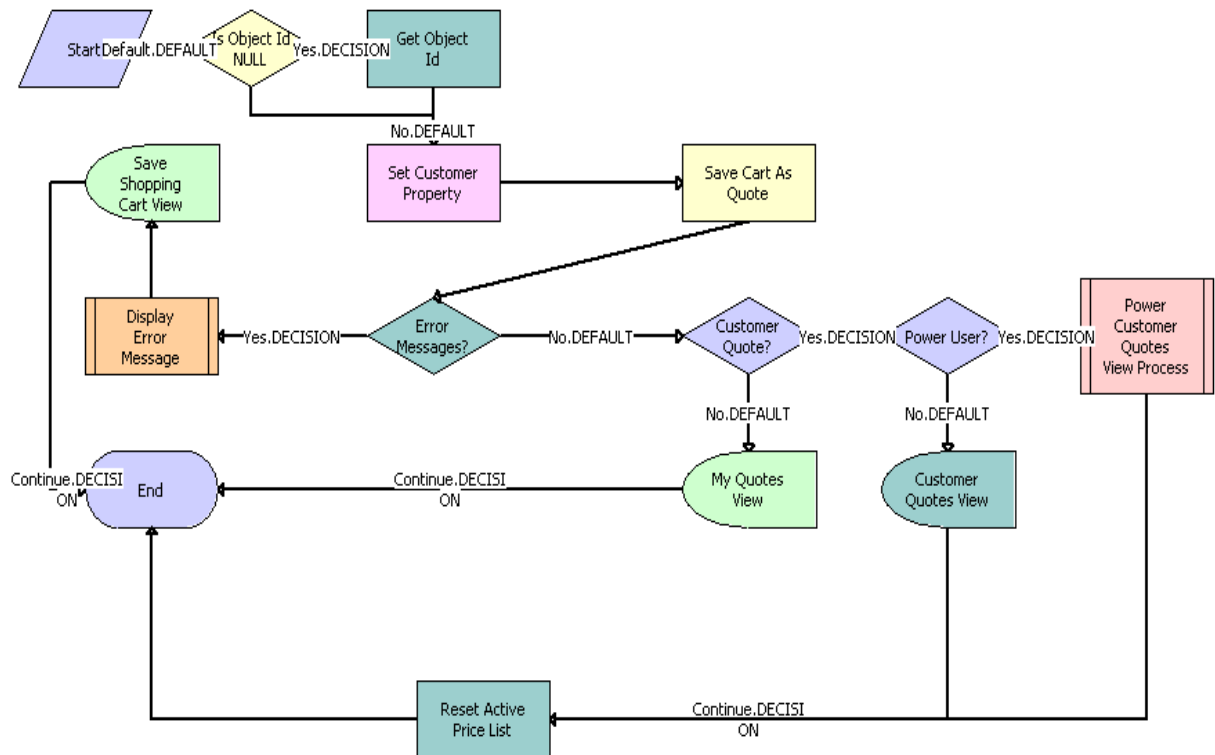


Figure 5. eSales - Save Cart Process

The eSales - Save Cart Process is called when a user clicks Save Cart in the Save Shopping Cart view. The following events happen:

- When an anonymous user with no associated quote logs in, the process obtains the value of the CurrentQuoteId profile attribute and uses that quote when saving the cart.
- The Save Shopping Cart view appears, where the user has two choices:
  - **Cancel.** The user is returned to the Shopping Cart.



- **Save As Quote.** The Shopping Cart is saved as a quote. The process checks for error messages.

If there is an error message, the eSales - Display Error Process (see [Figure 32 on page 104](#)) is called, and the user is sent back to the Save Shopping Cart view. This would happen, for example, if the user tried to save a quote with a name that was already being used.

If there is no error message, the process determines whether the quote is for an existing customer. If it is not, the My Quotes view appears. If it is, the process determines whether the customer is a power user.

**NOTE:** A power user is a delegated customer administrator or purchasing manager.

- If the customer is not a power user, the Customer Quotes View appears.
- If the customer is a power user, the eSales - Power Customer Quotes View process is called.

## eSales - Add To Template Process

Customers create templates to store groups of products that they may want to buy in the future. The eSales - Add To Template workflow is used to add:

- The entire contents of the shopping cart to a template.
- Individual products from a catalog to a template.

The workflow is shown in [Figure 6](#).

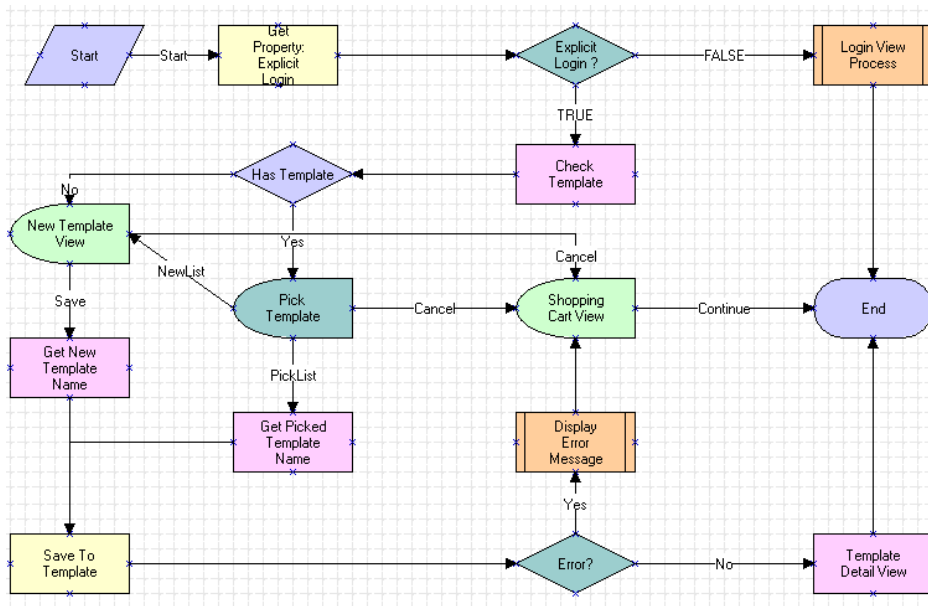


Figure 6. eSales - Add To Template Process

The eSales - Add To Template Process is called when a user clicks the Add To Template button in the shopping cart or catalog product detail view. The following events happen:

- The process establishes whether the customer has explicitly logged in. If the customer has not logged in, the eSales - Login View process is called to allow the customer to log in. If the customer has logged in, the process checks whether the customer has one or more existing templates. If the customer:
  - Has one or more templates, the Pick Quote Template View (eSales) appears and the customer either picks a template or chooses to create a new template.
  - Chooses to create a new template, or does not have a template, the New Quote Template View (eSales) appears and the customer creates a template.
- If the workflow was launched from the Shopping Cart, the line items from the Shopping Cart are added to the template. If the workflow was launched from a product catalog, the selected product is added to the template.
- The template is saved. If there is an error, an error message appears and the Shopping Cart View is returned.

## eSales - Delete Old Anonymous Quotes

The workflow for deleting old anonymous quotes is shown in [Figure 7](#).



Figure 7. eSales - Delete Old Anonymous Quotes

The Delete Old Anonymous Quotes process is called by a repeating server request according to the rules that you set up. For more information, see [Chapter 6, "Administering Siebel eSales."](#)

## Siebel eSales Checkout Workflows

The following workflows are associated with checking out in Siebel eSales:

- [SS Start Checkout Process on page 76](#)
- [Credit Check - Quotes \(eSales\) on page 77](#)
- [eSales - Enter Shipping Details Process on page 78](#)
- [SS Save Shipping Address Details Process on page 80](#)
- [eSales - Enter Payment Details Process on page 82](#)

- SS Save Payment Details Process on page 84
- eSales - Complete Checkout Process on page 86
- eSales - Authorize Order Process on page 87
- SS Authorize Order Process on page 89
- eSales - Save Credit Card on page 90
- SS Set Shipping Checkout Mode Process on page 91
- eSales - Save Shipping Address on page 92
- eSales - Save Account Shipping Address on page 92
- eSales - Set Primary Account Address on page 93
- eSales - Setting PO Status on page 94
- eSales - Current User Type on page 95
- SS Change Checkout Contact Detail Process on page 95
- SS Cancel Checkout Process on page 96

## SS Start Checkout Process

The workflow that is called when a user clicks on checkout from the shopping cart is shown in Figure 8. The user is then redirected to single page checkout view from where he can proceed to select shipping and payment details.

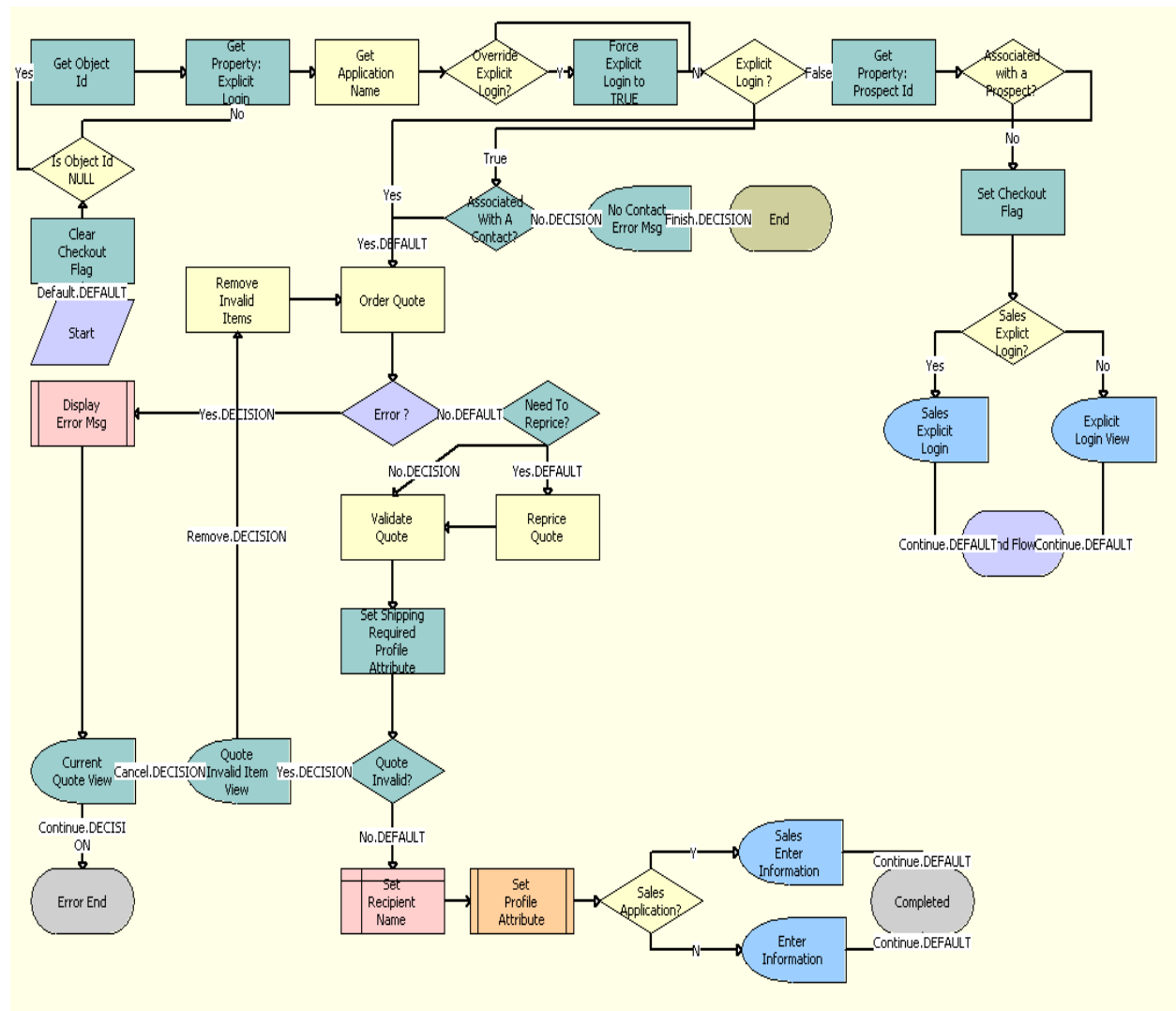


Figure 8. SS Start Checkout Process

The following events happen:

- The process reprices the product based on the pricelist and validates the quote.
- An error message is displayed if any of the following events occur:
  - If there are any invalid items in the cart.

- If there is no contact selected for the cart.
- If the cart is empty.
- If there are no errors, then the Checkout view is displayed. The user is then prompted to provide shipping and payment details to complete the checkout process.

## Credit Check - Quotes (eSales)

The workflow for performing a credit check is shown in Figure 9.

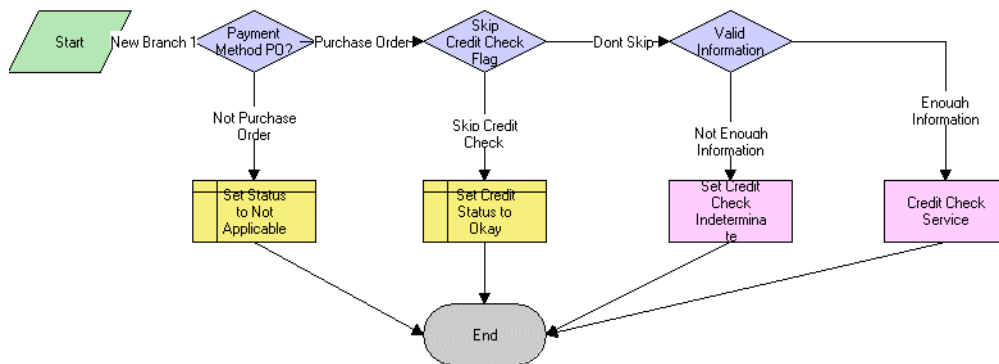


Figure 9. Credit Check Process

The Credit Check process is called from the eSales - Authorize Order Process. The following events happen:

- The process checks that the payment will be made with a PO. If it is not, the process sets the status of the credit check to Not Applicable.
- For orders that will be paid for using a PO, the process determines whether the Skip Credit Check check box is selected for the ordering account. If it is, the credit check is skipped. If it is not, the process determines whether the amount of the order is below the amount of the credit check threshold for the account (if one is set). If it is, the credit check is skipped. If the credit check is skipped, the Credit Status for the order is set to approved.
- The process checks that the billing information is complete. If not, the process sets the credit check status to Indeterminate. If there is enough information, the credit check is performed.



- If necessary, the process determines whether the customer has a User Type value of Web Registered User. Web Registered User users can choose personal shipping addresses from their address books or enter new shipping addresses. Users that do not have the Web Registered User Type choose account shipping addresses. If the account has no associated address, the number of account addresses is counted. If the address book is empty, the Account Shipping View appears, allowing the customer to enter an address. If there is more than one address, the Select Account Address View appears, allowing the customer to select a shipping address.
- The process determines whether the address needs to be validated. For certain applications, such as Siebel eEvents Management, Siebel Marketing, and Siebel Sales, it does not need to be validated. If necessary, the process validates the address selected or entered; that is, it checks to see that the required fields are completed. If it is invalid, an error message appears to the user and the process starts over again.
- For customers who are associated with accounts, the process determines whether the account address should be saved and calls the Save Account Shipping Address subprocess (see [Figure 20 on page 92](#)), followed by the eSales - Set Primary Account Address subprocess (see [Figure 21 on page 93](#)). The latter process sets a primary address the first time an account user checks out.
- For customers who are not associated with accounts, the eSales - Save Shipping Address workflow (see [Figure 19 on page 92](#)) is called.

## SS Save Shipping Address Details Process

The workflow for validating and saving the address selected by the customer to the Quote is shown in Figure 11.

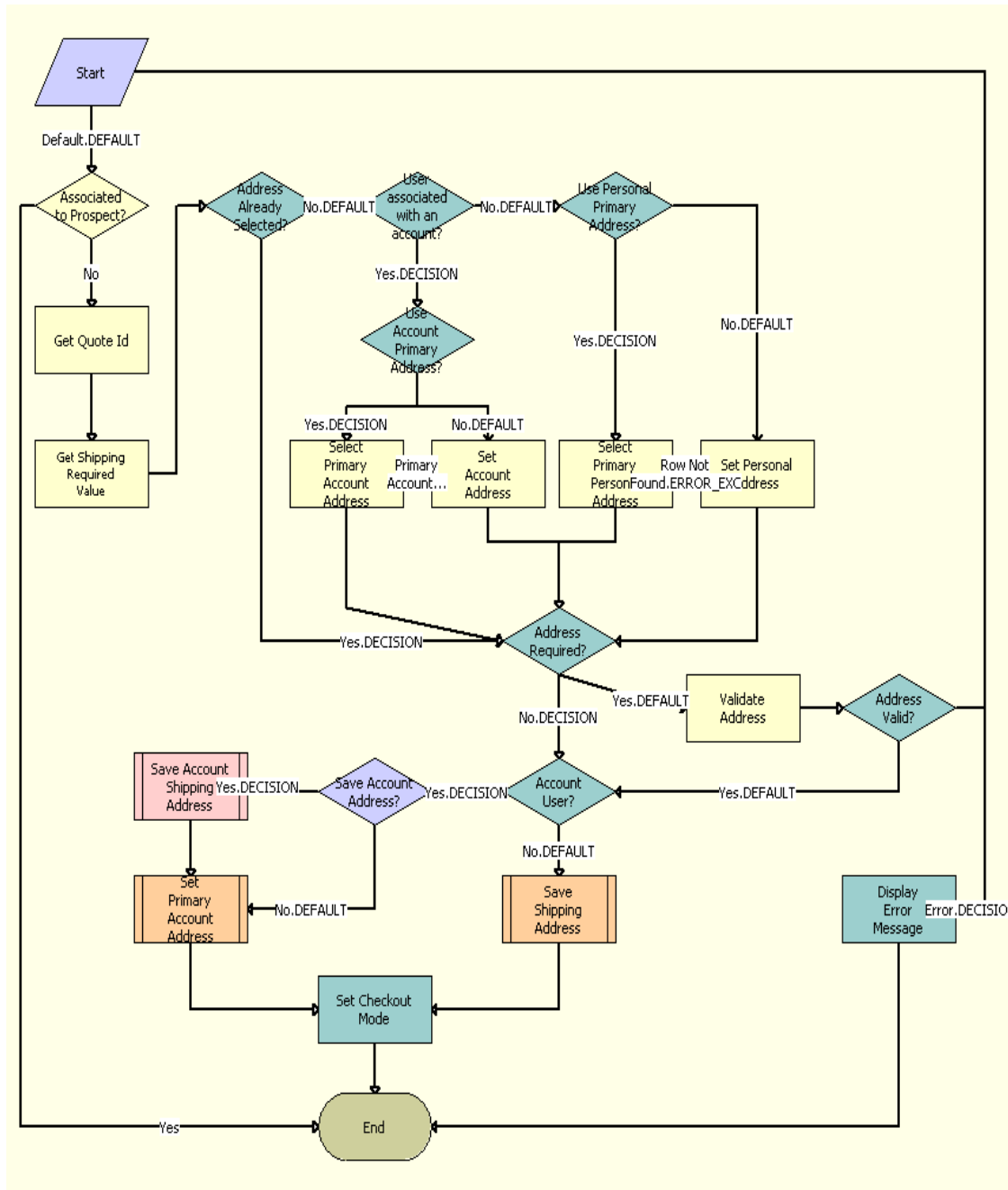


Figure 11. SS Save Shipping Address Details Process



The SS Save Shipping Address Details Process is called when the user selects a Shipping Address and clicks on Continue. The following events happen:

- The process identifies the address selected by the customer and associates it to the quote. If the below conditions are true, then the process associates the selected address to Account Ship To Street Address. If these conditions are not true, then the process associates the selected address to Personal Ship To Street Address.
  - If the quote has a valid Account.
  - If the customer is not one of the following user types:
    - Web Corporate User
    - Web Delegated Customer Administrator
    - Web Purchasing Manager
- The process determines if the selected address must be validated. If the address should be validated, then the process checks if the required fields are populated in the selected address. If the address is invalid, then an error message is displayed to the customer. The customer must then select a valid address or edit the fields in the selected address.
- The process checks the following conditions:
  - If the Quote has a valid account
  - If the customer is not one of the following user types.
    - Web Corporate User
    - Web Delegated Customer Administrator
    - Web Purchasing Manager
  - If these conditions are true, then the process calls the eSales - Save Account Shipping Address subprocess (see [Figure 20](#)) to save the address. The process then calls the eSales - Set Primary Account Address subprocess (see [Figure 21](#)) to set a primary address when an account user checks out for the first time.
- If these conditions are not true, the process calls the the eSales - Save Shipping Address subprocess (see [Figure 19](#)) to complete the following actions:
  - Save the address.
  - Set to the primary address when the contact user checks out for the first time.
  - Associate the address to the contact.
- The process sets the value of the SSCheckoutViewMode profile attribute to Mode-3 so that the customer can choose the payment method.

## eSales - Enter Payment Details Process

The workflow for selecting, entering, and modifying payment information is shown in Figure 12.

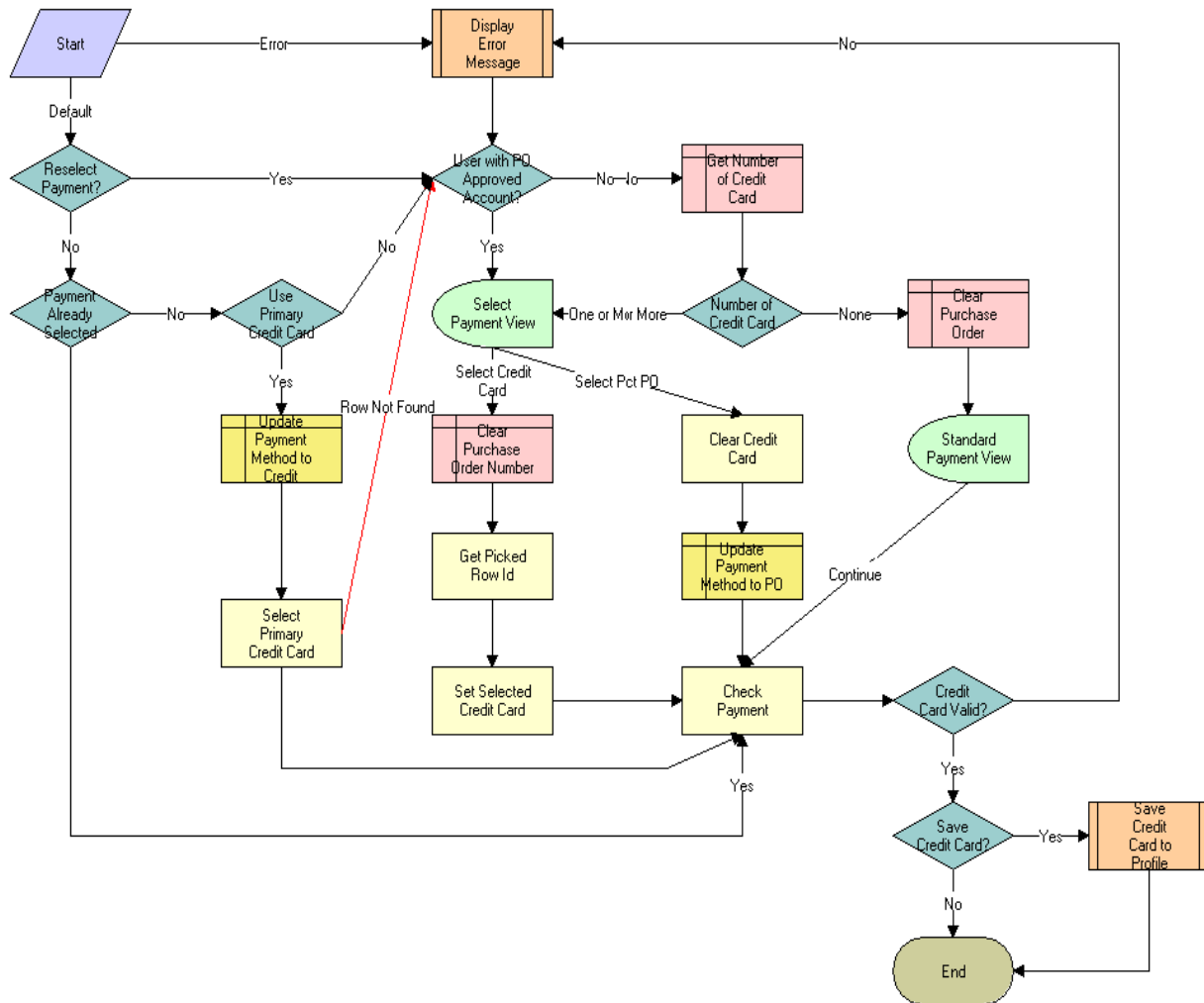


Figure 12. eSales - Enter Payment Details Process

When payment information needs to be entered, the eSales - Enter Payment Details Process is called. The following events happen:

- If there is an error in the workflow, the eSales - Display Error Process (see [Figure 32 on page 104](#)) workflow is called.
- If there is no error, the process determines whether payment has already been selected (that is, it has been called from the Order Summary view) or not (that is, it has been called from the Shopping Cart).

- If the process was called from the Shopping Cart, it determines whether a payment method has already been selected (for example, if the user previously began the checkout process but did not complete it).
  - If the payment method has already been selected, the process skips to the validation steps.
  - If not, the user must select or enter payment information.
- If the process was called from the Order Summary view, the user must reselect or re-enter payment information.
- If necessary, the process determines whether the customer is a B2B user associated with an account approved for using purchase orders (POs). B2B customers with PO accounts are sent directly to the Select Payment view.

For customers without PO accounts, the process determines whether they have credit card information saved in their profiles.

- If no credit card information exists, the customer must enter payment information that the process validates.
- If credit card information exists, the customer is sent to the Select Payment view.
- In the Select Payment view, PO-approved B2B customers choose either PO or credit card. Customers who are not approved to pay by PO can pay with a credit card by selecting an existing card or entering a new one.
- If payment is made by credit card, the credit card is validated. Validation consists of checking that the credit card number is valid (using the Mod 10 algorithm), that the expiration date is in the future, and that all required fields are completed.

If the number is not valid, an error message appears and the payment process is restarted.

- New credit card numbers can be added to user profiles if users indicate that the numbers should be stored for future use.

## SS Save Payment Details Process

The workflow for selecting, entering, and modifying of payment information is shown in Figure 13.

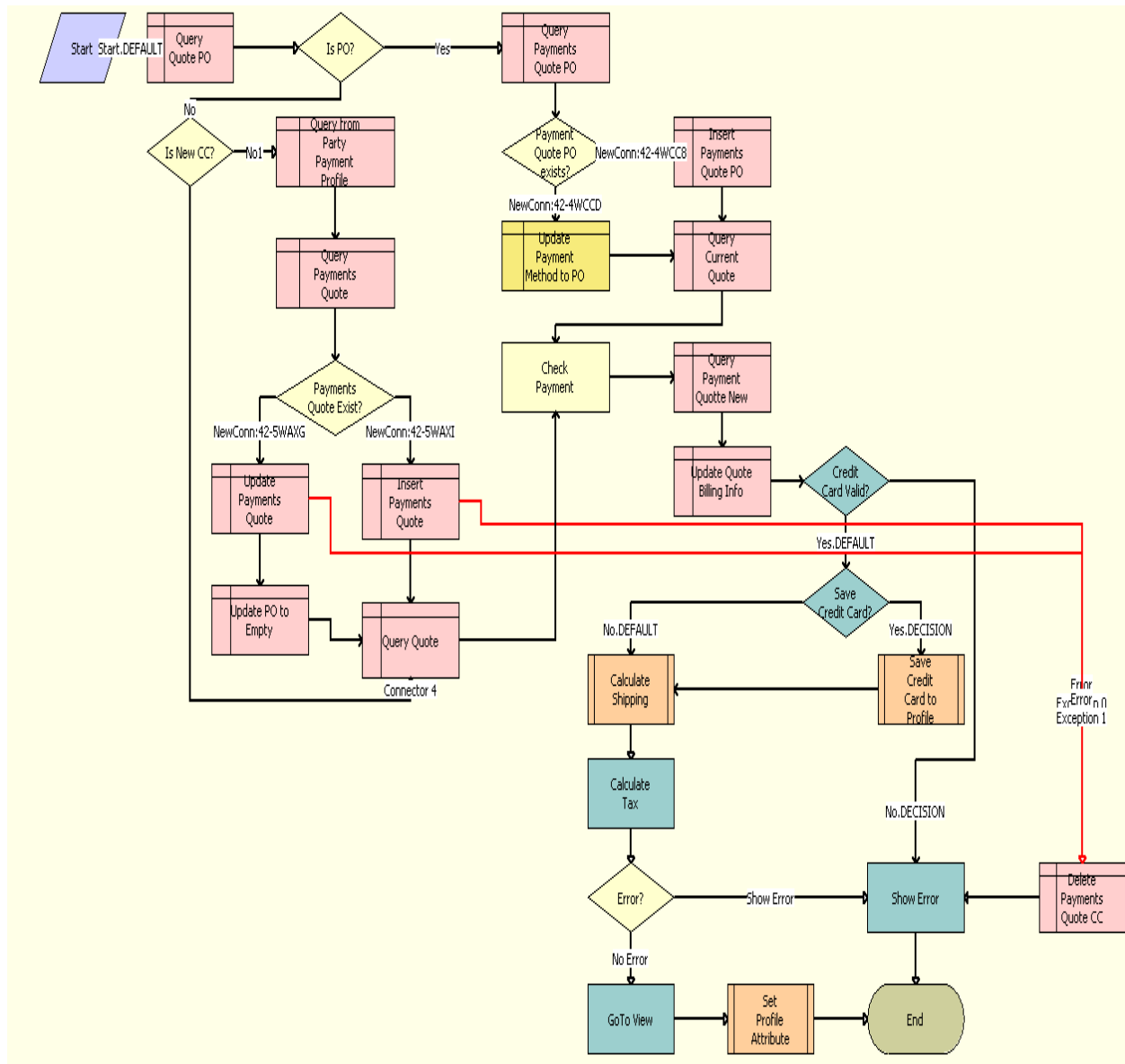


Figure 13. SS Save Payment Details Process

When the user clicks Continue in the Payment applet of the Checkout View, the SS Save Payment Details Process is called. The following events happen:

- The process checks if the customer entered a purchase order number. If a purchase order number is available, the following events happen:

- The process checks if there is payment information in the profile.
- If there is payment information available, then the process updates the profile with the current purchase order details
- If there is no payment information available, then the process creates a new entry in the profile with the current information.
- If there is no purchase order number available, then the process checks if the customer is paying with a new credit card.
  - If the customer is not paying with a new credit card, then the process picks the selected credit card in Credit Cards Applet of the Checkout View. The process updates the Payment Profile with the selected Credit Card details.
  - If an error occurs when the credit card details are added, then the process deletes the card details and displays an error to the customer.
- The process completes the following actions:
  - Validates the payment .
  - Updates the billing information.
  - Calculates the shipping cost.
  - Calculates any applicable tax.
  - Checks for errors. If an error occurs, then an error message is displayed to the customer and the process ends.
- The Check Out View is displayed.

## eSales - Complete Checkout Process

The workflow for checkout completion is shown in Figure 14.

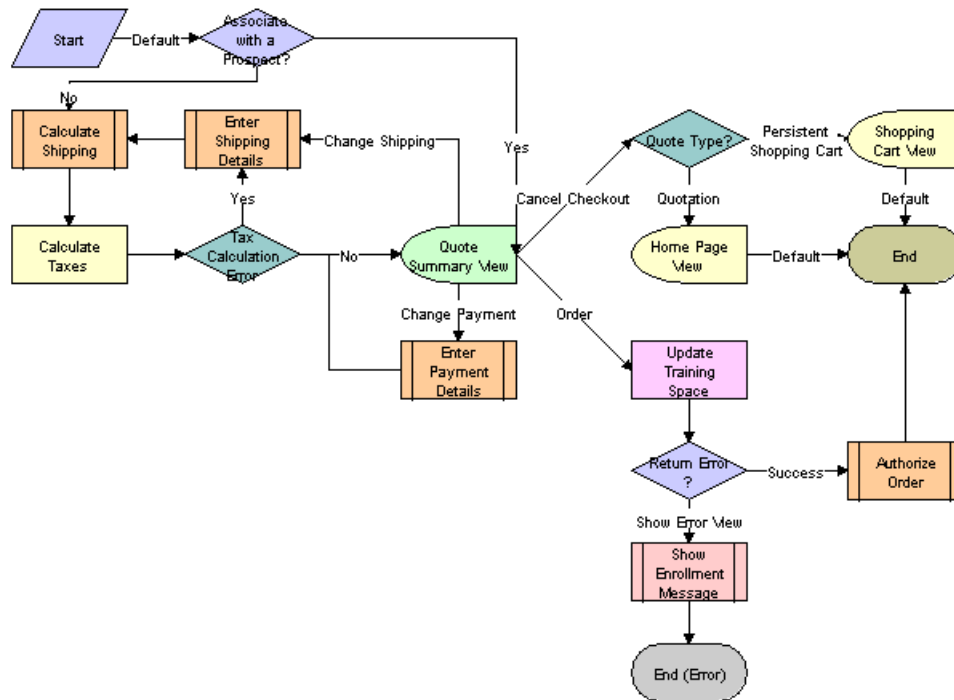


Figure 14. eSales - Complete Checkout Process

In the eSales - Checkout Process, after shipping and payment information have been added to the order, the eSales - Complete Checkout Process is called. The following events happen:

- Shipping charges are calculated using the Shipping Cost Service (eScript) business service.
- Sales taxes are calculated using the Tax Calculator and Taxware Adapter Service business services. For more information about tax calculation, see the section about third-party taxation applications in *Siebel Order Management Guide*.

If there is a tax calculation error, the customer must enter shipping address information again. This is because tax rates depend on having a valid shipping address.

If non-ASCII characters are entered in any field that is passed to Taxware, the process checks whether the field is required. If it is required, the process sets the error message into the quote header and copies it to the order. The customer sees an error message stating that the information required to calculate taxes could not be processed but that the order will be placed and the customer will receive an email with the total cost of the order.

The customer can then confirm the order. The process copies the error message to the order so that an administrator can query for the order with the problem.

- The customer is sent to the Quote Summary view, where the customer can edit shipping and payment information, cancel the checkout process, or confirm the order.

- Entering new shipping information causes shipping charges and taxes to be recalculated.
- Choosing to cancel checks the quote type. If Quotation, the customer is sent to the home page. If Persistent Cart, the customer is sent to the Shopping Cart. For more information on quote types, see [“Quote Types in Siebel eSales” on page 57](#).
- Choosing to confirm the order calls the eSales - Authorize Order Process (see [Figure 15 on page 87](#)).

## eSales - Authorize Order Process

The workflow for authorizing credit card payments and purchase orders is shown in [Figure 15](#).

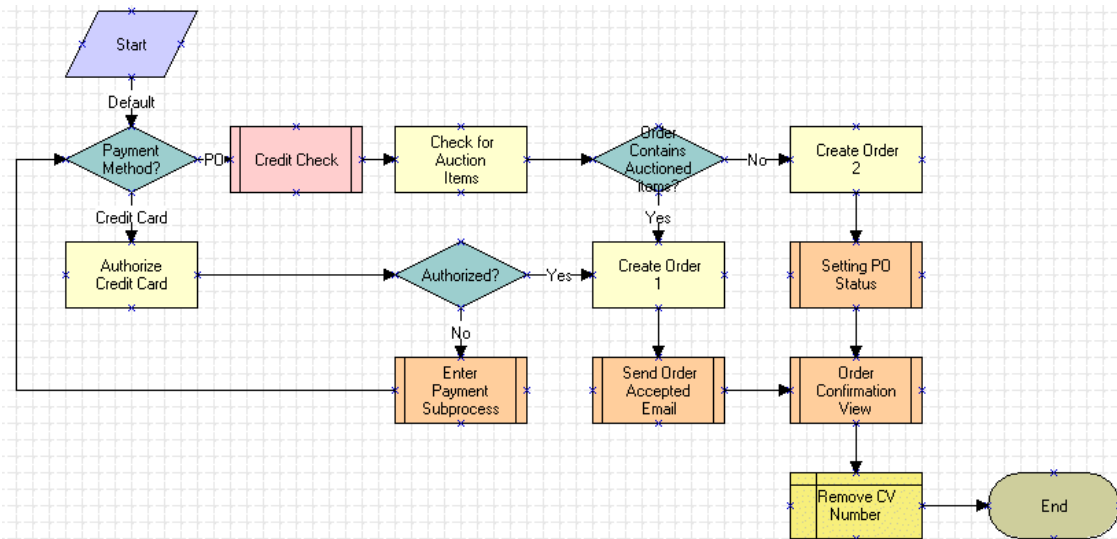


Figure 15. eSales - Authorize Order Process

Confirming the order in the Order Summary view calls the eSales - Authorize Order Process. The following events happen:

- The process determines which payment method was selected, PO or credit card.
- If the PO method was selected, the process:
  - Performs a credit check by calling the Credit Check - Quotes (eSales) process. If the credit check fails, the order is still created, but the Credit Status is set to a value such as Declined or Over Limit.
  - Checks for auction items in the order.

**NOTE:** Auctions are not part of the current Siebel Business Applications functionality. This step is included for backward compatibility only.

- If there are none, the purchase order is created using the CreateOrder method of the Shopping Service business service, the Setting PO Status workflow is called (see [Figure 22 on page 94](#)), and then the Order Confirmation view is shown using the eSales - Order Confirmation View process (see [Figure 34 on page 106](#)).
- If there are auction items, a PO cannot be used. This is because once an auction has been won, it cannot be rejected by a purchasing manager. The order is created, an order acceptance email is sent, the Order Confirmation view is shown, and the Credit Card Verification Number is deleted.
- If a credit card was used, the credit card is authorized using the workflow Credit Card Transaction Service.
  - If authorization is successful, an order is created using the CreateOrder method of the Shopping Service business service. An order acceptance email is sent, the Order Confirmation view appears, and the Credit Card Verification Number is deleted.
  - If authorization fails, the eSales - Enter Payment Details Process (see [Figure 12 on page 82](#)) is called. Credit card authorization is performed again and if it is successful, the order is created.

See [Chapter 4, "Integrating Siebel eSales with Third-Party Payment Applications,"](#) for more information on credit card authorization.



## SS Authorize Order Process

The workflow for creating a sales order for a specific quote is shown in [Figure 16](#).

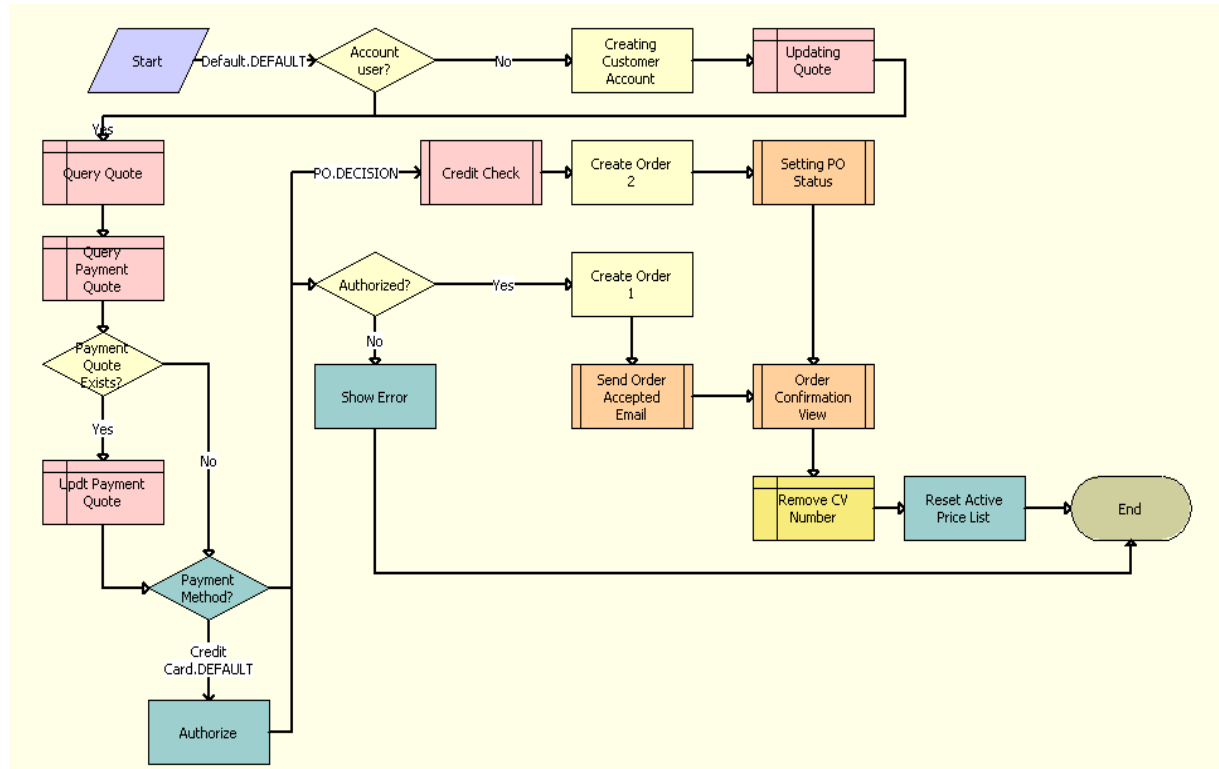


Figure 16. SS Authorize Order Process

This workflow also performs a credit card authorization where a credit card is used, and where a credit check is required in a purchase order. The following events happen:

- If there is no account ID, then the process creates a dummy customer account.
- The process updates the billing account ID with the dummy customer account details.
- The process queries the current quote and retrieves the quote total amount used to update the payments step.
- The process updates the payment quote record to match the quote total amount.
- If a credit card was used for payment, then the process authorizes the payment and the following events happen:
  - If the credit card authorization fails, then an error message is displayed and the process ends.
  - If the credit card payment is authorized, then a sales order is created for the current quote.
- The process calls the eSales - Send Order Accepted Email workflow (see [“Siebel eSales Notification Workflows” on page 96](#)), which sends an order acceptance email to the customer.

- The process calls the SS Order Confirmation View Process (see [Figure 35](#)) to navigate to the Order Confirmation view.
- For purchase orders, the process calls the eSales - Credit Check - Quotes (see [Figure 9](#)) to perform a credit check.
- For purchase order flows, a sales order is created for the current quote.
- For purchase orders, the process calls the eSales - Setting PO Status workflow (see [Figure 22](#)) to validate the purchase order, to set the order status, and to send the order acceptance mail.
- The process removes the CCV number from the quote.

## eSales - Save Credit Card

The workflow for saving credit card information is shown in [Figure 17](#).

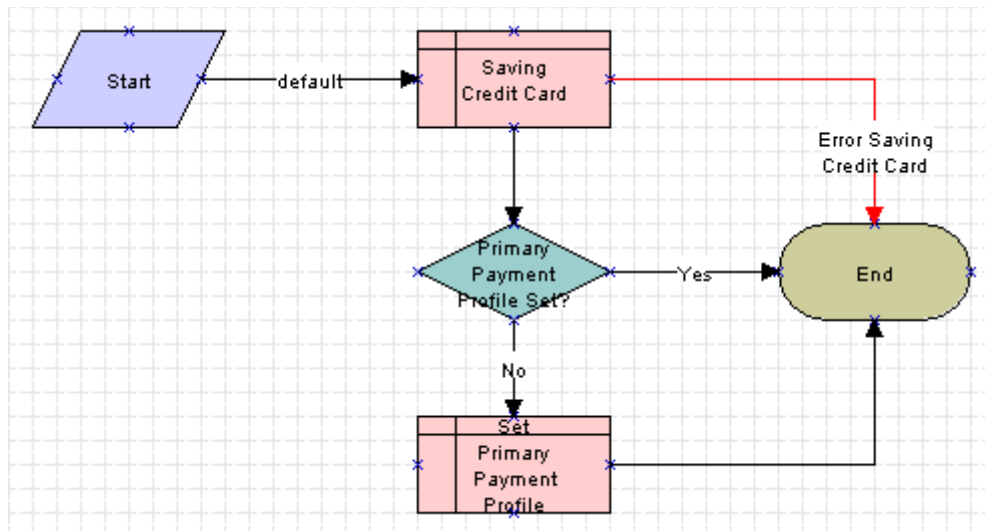


Figure 17. eSales - Save Credit Card

When the eSales - Save Credit Card process is called by the eSales - Enter Payment Details Process (see [Figure 12 on page 82](#)), the following events happen:

- The process saves the credit card information to the Personal Payment Profile business component.
- The process determines whether the user has a primary credit card already set.
  - If so, the process ends.
  - If not, the current credit card is set as the primary.
- Any error generates an exception.

## SS Set Shipping Checkout Mode Process

The workflow that decides if an account's or contact's shipping address is displayed to the user is shown in [Figure 18](#).

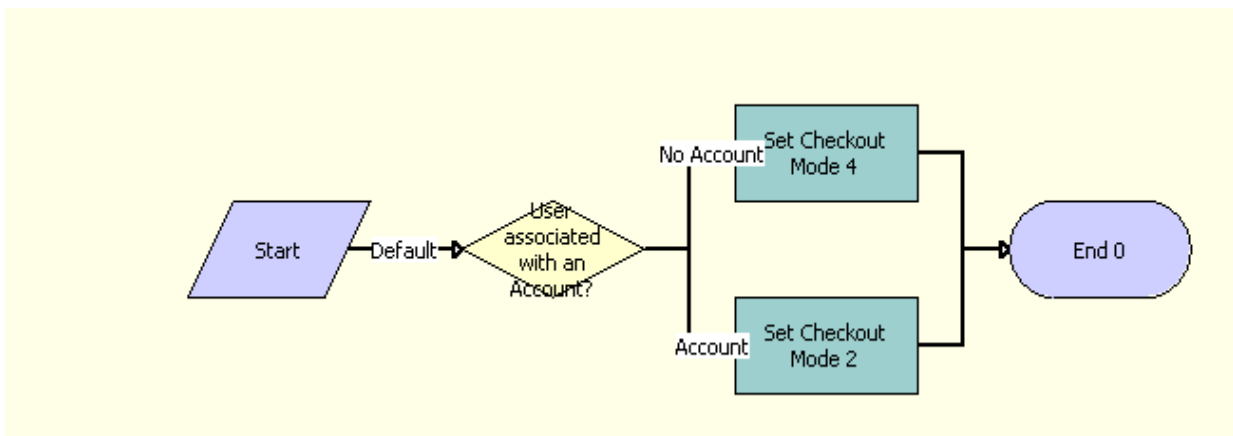


Figure 18. SS Set Shipping Checkout Mode Process

The SS Set Shipping Checkout Mode Process is called when the user performs one of the following actions:

- Clicks Checkout in the Shopping Cart View.
- Clicks Continue in the Contact List applet in the Checkout View
- Clicks Edit in the Shopping Form applet in the Checkout View
- Clicks Continue in the Credit Card/Purchase Order applet in the Checkout View

The following events happen:

- The process checks the following conditions:
  - If the quote has a valid account.
  - If the customer is not one of the following user types:
    - Web Corporate User
    - Web Delegated Customer Administrator
    - Web Purchasing Manager
- If these conditions are true, then the process displays the account's shipping address.
- If these conditions are not true, then the process displays the contact's shipping address.

## eSales - Save Shipping Address

The workflow for saving shipping address information is shown in [Figure 19](#).

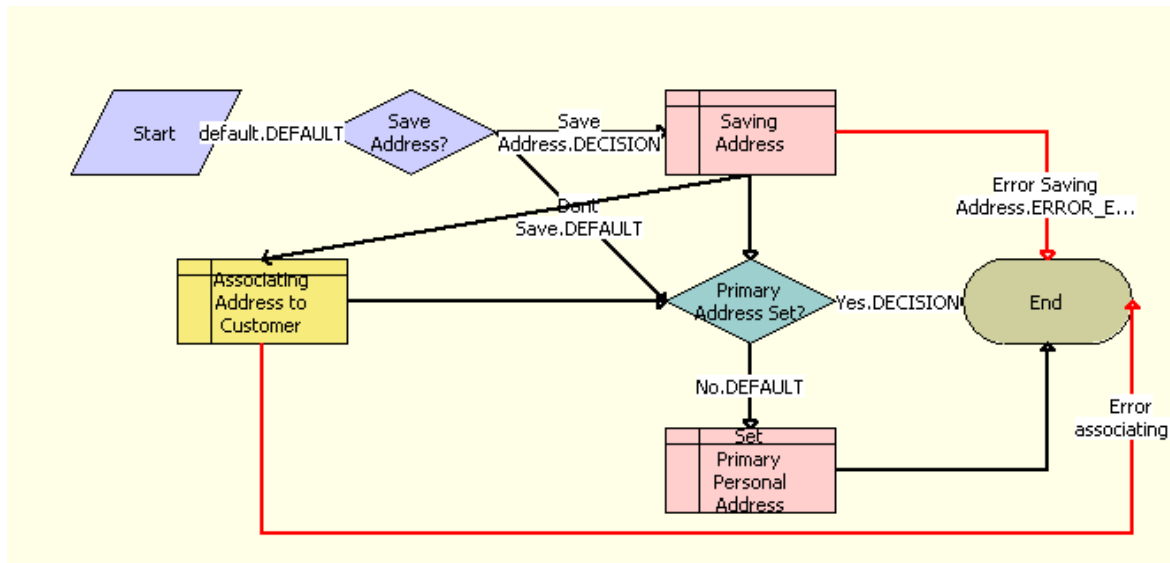


Figure 19. eSales - Save Shipping Address

When the eSales - Save Shipping Address workflow is called by the SS Save Shipping Details Process (see [Figure 11 on page 80](#)), the following events happen:

- The process determines whether the shipping address needs to be saved. If it does, it saves the shipping address information to the Personal Address business component.
- The process associates the shipping address to the contact.
- The process determines if the contact has a primary address. If the contact has no primary address, then the address that the user selects becomes the primary personal address.

## eSales - Save Account Shipping Address

The workflow for saving the account shipping address is shown in [Figure 20](#).

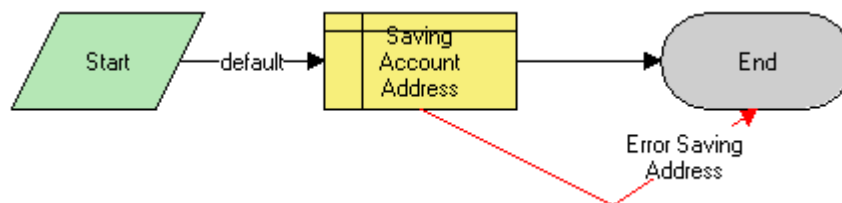


Figure 20. eSales - Save Account Shipping Address

When the eSales - Save Account Shipping Address workflow is called by the SS Save Shipping Details Process (see [Figure 11 on page 80](#)), it saves the account shipping address to the CUT Address business component.

## eSales - Set Primary Account Address

The workflow for setting the primary account address is shown in [Figure 21](#).

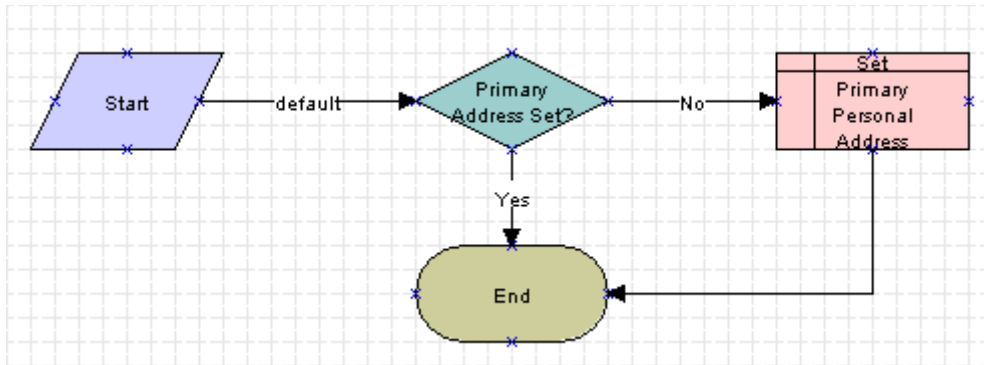


Figure 21. eSales - Set Primary Account Address

When the eSales - Set Primary Account Address workflow is called by the SS Save Shipping Details Process (see [Figure 11 on page 80](#)), it determines whether the B2B user has a primary account address.

- If so, nothing is done.
- If not, the current address is saved to the User Profile (eApps) business component as the primary address for that B2B user.

## eSales - Setting PO Status

The workflow for setting the status of a purchase order is shown in [Figure 22](#).

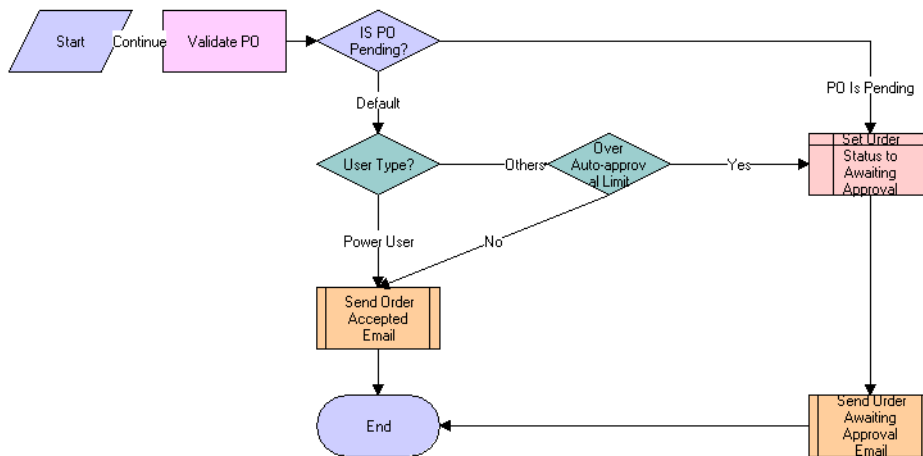


Figure 22. eSales - Setting PO Status

When the eSales - Setting PO Status workflow is called by the eSales - Authorize Order Process (see [Figure 15 on page 87](#)), the following events happen:

- The process determines whether the PO is pending.
  - If it is, the order status is set to Awaiting Approval and the Send Order Awaiting Approval Email (eSales) process is called.
  - If it is not, the process determines the user type.
- If the user is a power user, the Send Order Accepted Email (eSales) process (see [“Siebel eSales Notification Workflows” on page 96](#)) is called.
- If the user is of another type, the process determines whether the order amount is over the auto-approval limit.
  - If it is not over the auto-approval limit, the Send Order Accepted Email (eSales) process is called.
  - If it is over the limit, the order status is set to Awaiting Approval and the Send Order Awaiting Approval Email (eSales) process is called.

For more information about PO status, see [Chapter 6, “Administering Siebel eSales.”](#)

## eSales - Current User Type

The workflow for looking up the current user type from the contact record is shown in [Figure 23](#).

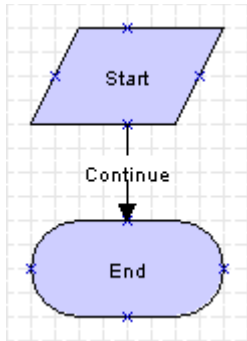


Figure 23. eSales - Current User Type

The eSales - Current User Type workflow is called by the eSales - Approve Order Process (see [Figure 27 on page 98](#)) and the eSales - Reject Order Process (see [Figure 28 on page 99](#)).

## SS Change Checkout Contact Detail Process

The workflow for changing the contact details during checkout is shown in [Figure 24](#).

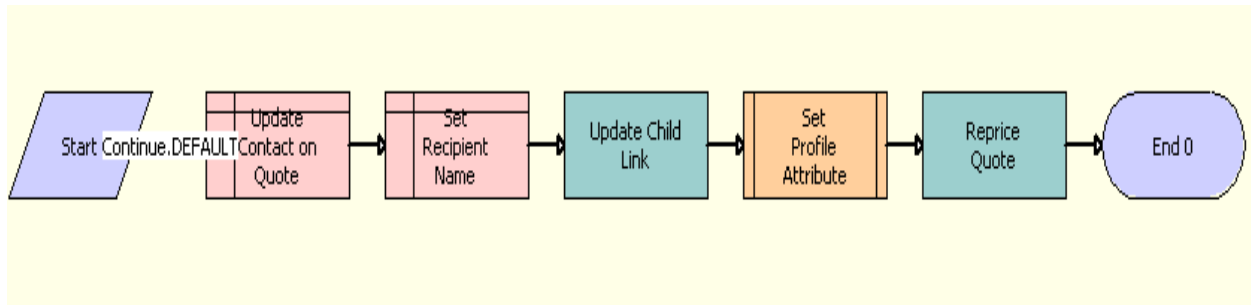


Figure 24. SS Change Checkout Contact Detail Process

When the SS Change Checkout Contact Detail workflow is called, the following happens:

- The process updates the selected contact on the quote and sets the contact name in the Recipient Name field.
- Based on the selected contact name, the process updates the shipping and payment records.
- The process calls SS Set Shipping Checkout Mode Process (see [Figure 18 on page 91](#)) to set the mode for the checkout view and then reprices the quote based on the selected contact name.

## SS Cancel Checkout Process

The workflow for cancelling the checkout is shown in [Figure 25](#).

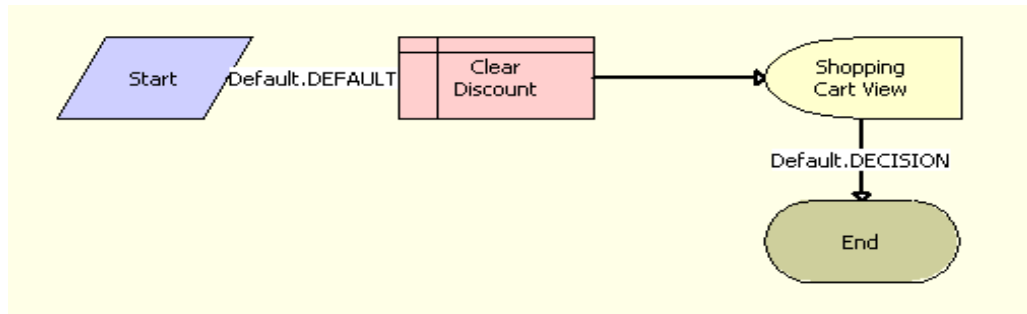


Figure 25. SS Cancel Checkout Process

When the SS Cancel Checkout Detail Process workflow is called, the process clears any discount applied to the quote and returns to the Shopping Cart view.

## Siebel eSales Notification Workflows

The following workflows send email to users about the status of their orders:

- **Send Order Accepted Email (eSales)**. This workflow notifies all users who place an order successfully.
- **Send Order Awaiting Approval Email (eSales)**. This workflow notifies the purchasing manager for an account when an order with a status of Awaiting Approval needs to be reviewed.
- **Send Order Failed Email (eSales)**. This workflow notifies a user when the credit card used cannot be authorized.
- **Send Order Rejected Email (eSales)**. This workflow notifies a corporate user when the purchasing manager or delegated customer administrator has rejected that corporate user's order.
- **Send Order Shipped Email (eSales)**. This workflow notifies a user when that user's order has been shipped.

This workflow is triggered by the Send Order Shipped Email (eSales) workflow policy action, which is triggered by the Send Order Shipped Email Policy (eSales) workflow policy when the order status is changed to Shipped.



All five processes take the form shown in [Figure 26](#). The process sends an email or an error generates an exception.

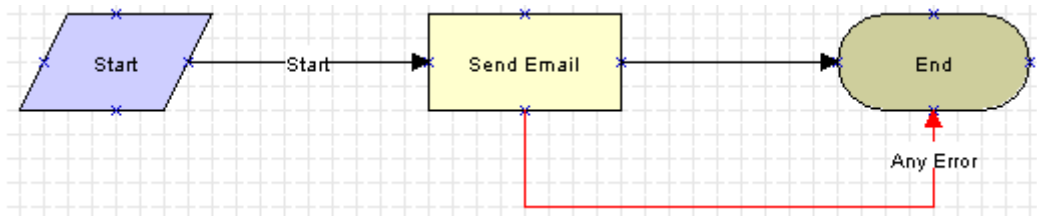


Figure 26. Email Notification Workflow

For more information on email notifications, see [“Customizing Email Notifications for Siebel eSales” on page 25](#). For information on workflow policies and workflow policy actions, see *Siebel Business Process Designer Administration Guide*.

## Siebel eSales Purchase Approval Workflows

A purchasing manager or a delegated customer administrator must approve or reject a purchase order if the order is over the auto-approval limit.

This section describes the following workflows:

- [eSales - Approve Order Process on page 98](#)
- [eSales - Reject Order Process on page 99](#)

## eSales - Approve Order Process

The workflow for approving purchase orders is shown in [Figure 27](#).

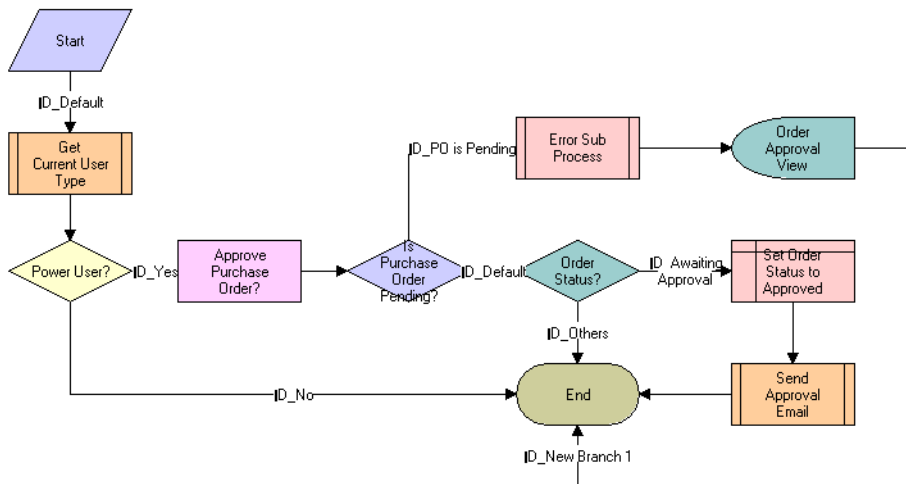


Figure 27. eSales - Approve Order Process

The eSales - Approve Order Process is called when a delegated customer administrator or purchasing manager clicks Approve in the My Orders view. The following events happen:

- The process calls the eSales - Current User Type workflow (see [Figure 23 on page 95](#)) to determine the user type.
- If the customer is not a power user, the process ends.
- If the customer is a power user, the process determines whether the PO status is Pending because no PO number was entered. If the PO status is Pending, the eSales - Display Error Process workflow is called, an error message is generated, and the customer is returned to the Order Approval view, where the customer can enter the PO number.
- If the PO status is not Pending, the process determines the order status.
  - If the order status is Awaiting Approval, the PO status is set to Approved and the Send Order Accepted Email (eSales) workflow (see [“Siebel eSales Notification Workflows” on page 96](#)) is called.
  - If the order status is not Awaiting Approval, the process ends.

## eSales - Reject Order Process

The workflow for rejecting purchase orders is shown in [Figure 28](#).

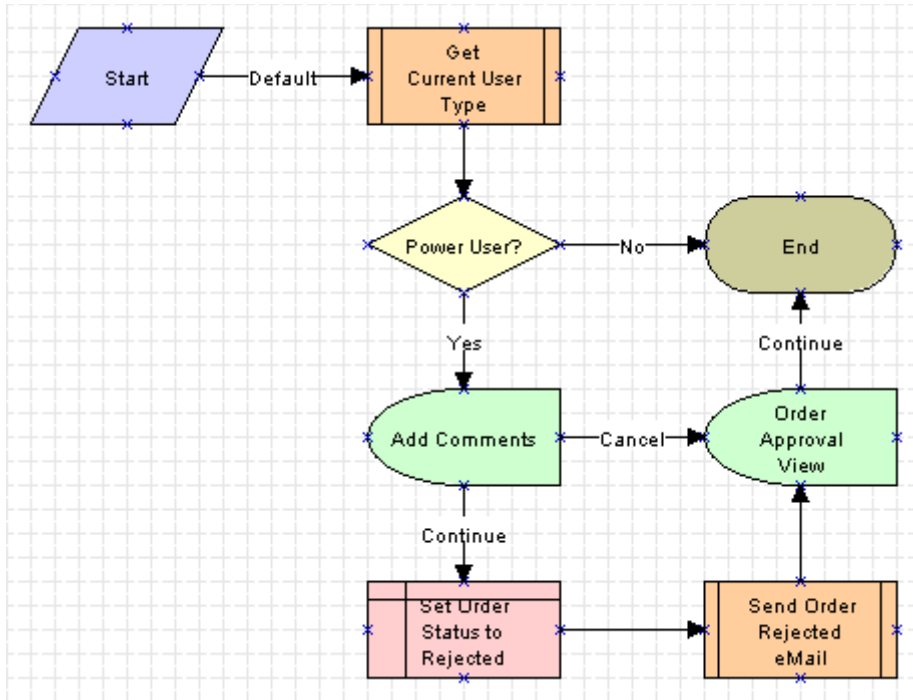


Figure 28. eSales - Reject Order Process

The eSales - Reject Order Process is called when a delegated customer administrator or purchasing manager clicks Reject in the My Orders view. The following events happen:

- The process calls the eSales - Current User Type workflow (see [Figure 23 on page 95](#)) to determine the user type.
- If the user is not a power user, the process ends.
- If the user is a power user, the Add Comments view appears, where the user can explain why the order has been rejected. The user then has two choices:
  - **Cancel.** The comments are not saved, and the Order Approval view appears. The status of the order is unchanged; it is still Awaiting Approval.
  - **Continue.** The comments are saved and the order status is set to Rejected. The Send Order Rejected Email (eSales) workflow (see ["Siebel eSales Notification Workflows" on page 96](#)) is called and the Order Approval view appears.

# Siebel eSales Order Modification Workflows

Orders can be canceled, modified, and reordered.

This section describes the following workflows:

- eSales - Cancel Order Process on page 100
- eSales - Modify Order Process on page 101
- eSales - Reorder Process on page 102

## eSales - Cancel Order Process

The workflow for canceling orders is shown in [Figure 29](#).

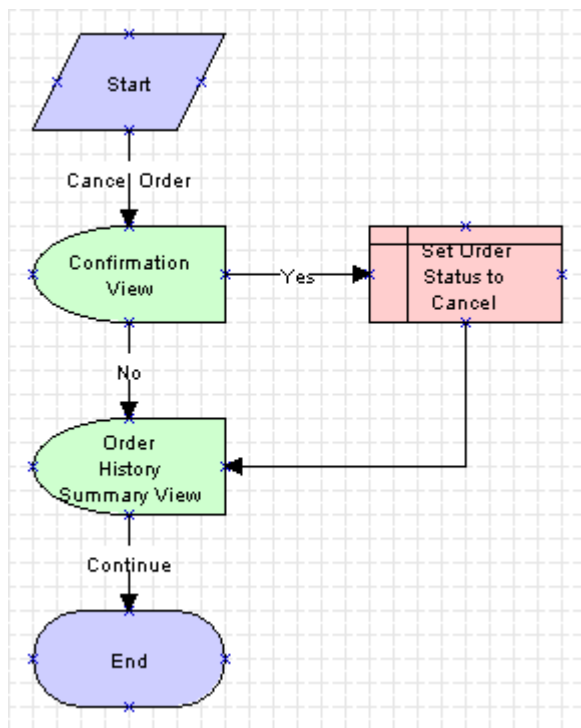


Figure 29. eSales - Cancel Order Process

The eSales - Cancel Order Process is called when a customer clicks Cancel in the Order History Summary view. The following events happen:

- The process displays the Cancel Order Confirmation view.

If the user confirms the cancellation, the order status is set to Canceled and the Order History Summary view appears.

If the user decides not to cancel the order, the user is returned to the Order History Summary view.

## eSales - Modify Order Process

The workflow for modifying orders is shown in Figure 30. Modifying an order cancels the original order and creates a new shopping cart with the same items in its place. The items in the shopping cart can then be modified and a new order created to replace the original order.

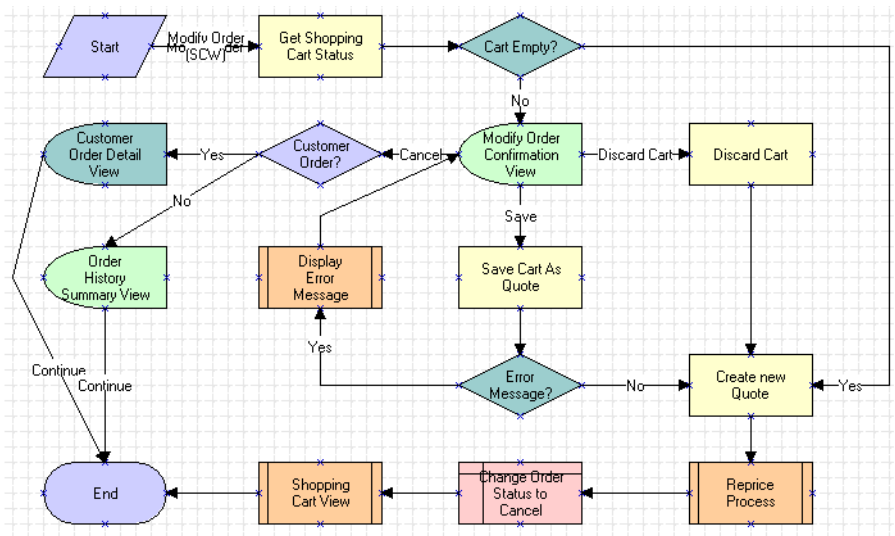


Figure 30. eSales - Modify Order Process

The eSales - Modify Order Process is called when a customer clicks Modify in the Order History Summary view. The following events happen:

- The process determines whether the Shopping Cart is empty.
- If it is empty, a new quote (Shopping Cart) is created from the order, the order is canceled, and the Shopping Cart appears.
- If it is not empty, the Modify Order Confirmation view appears, where the user has three choices:
  - **Cancel.** The process determines whether the quote is for an existing customer. If it is, the Customer Order Detail View appears and the process ends. If it is not, the Order History Summary View appears and the process ends.

- **Discard Cart.** The Shopping Cart is emptied and a new one is created from the order. The new Shopping Cart is repriced, the old order is canceled, and then the Shopping Cart appears. The eSales - Shopping Cart View process (see [Figure 36 on page 108](#)) is used to display the Shopping Cart.
- **Save Cart.** If there is an error, the eSales - Display Error Process (see [Figure 32 on page 104](#)) is called, and then the user is sent back to the Modify Order Confirmation view. For example, this would happen if the user tried to save a quote with a name that was already being used.

If there is no error, the Shopping Cart is saved as a quote (see ["eSales - Save Cart Process" on page 72](#)) and a new Shopping Cart is created from the order. The new Shopping Cart is repriced, the old order is canceled, and then the Shopping Cart appears. The eSales - Shopping Cart View process is used to display the Shopping Cart.

## eSales - Reorder Process

The workflow for reordering is shown in [Figure 31](#). It works similarly to modifying an order.

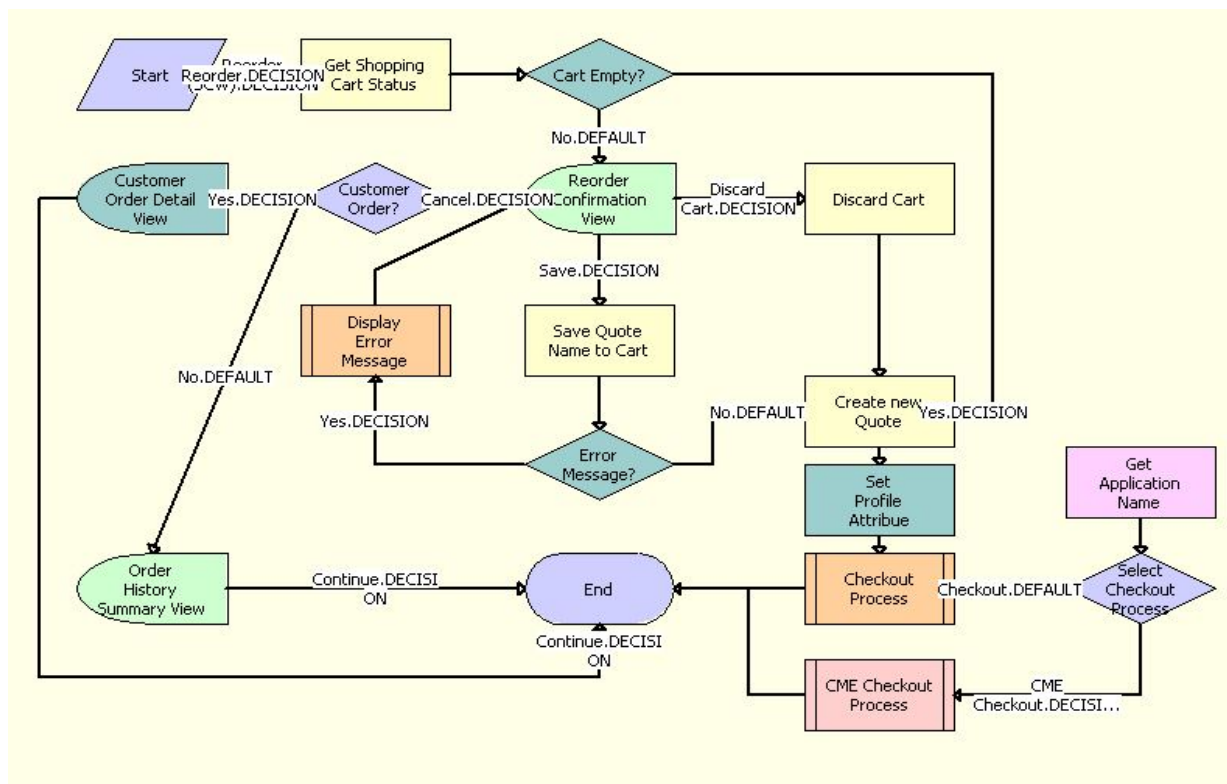


Figure 31. eSales - Reorder Process

The eSales - Reorder Process is called when a customer clicks Reorder in the Order History Summary view. The following events happen:

- The process determines whether the Shopping Cart is empty.
  - If it is empty, a new quote (Shopping Cart) is created from the order and the SS - Start Checkout Process (see [Figure on page 76](#)) is called.
  - If it is not empty, the Reorder Confirmation view appears, where the user has three choices:
    - **Cancel.** The process determines whether the quote is for an existing customer. If it is, the Customer Order Detail View appears and the process ends. If it is not, the Order History Summary View appears and the process ends.
    - **Discard Cart.** The Shopping Cart is emptied and a new Shopping Cart is created from the order. The eSales - Checkout Process is called.
    - **Save Cart.** If there is an error, the eSales - Display Error Process (see [Figure 32 on page 104](#)) is called, and then the user is sent back to the Reorder Confirmation view. For example, this would happen if the user tried to save a quote with a name that was already being used.
- If there is no error, the Shopping Cart is saved as a quote (see ["eSales - Save Cart Process" on page 72](#)) and a new Shopping Cart is created from the order. The SS - Start Checkout Process is called.

## Siebel eSales Error Message and View Display Workflows

The following processes display error messages and views that do not use the same business object as the workflow that called them.

This section describes the following workflows:

- [eSales - Display Error Process on page 103](#)
- [eSales - Login View on page 105](#)
- [eSales - Order Confirmation View on page 106](#)
- [SS Order Confirmation View Process on page 107](#)
- [eSales - Shopping Cart View on page 108](#)

### eSales - Display Error Process

The process used for displaying error messages is shown in [Figure 32](#). It is called by the following workflows:

- eSales - Edit Quote Process ([Figure 2 on page 68](#))
- eSales - Save Cart Process ([Figure 5 on page 72](#))
- eSales - Enter Shipping Details Process ([Figure 10 on page 78](#))
- eSales - Checkout Process ([Figure 8 on page 76](#))
- eSales - Enter Payment Details Process ([Figure 12 on page 82](#))

- eSales - Modify Order Process (Figure 30 on page 101)
- eSales - Reorder Process (Figure 31 on page 102)

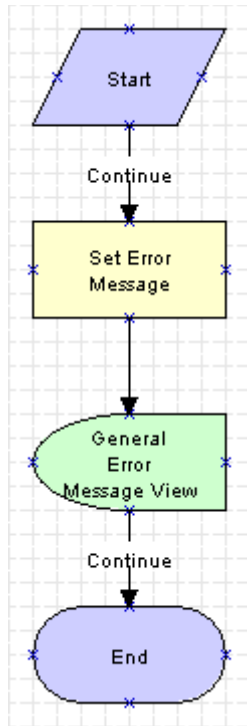


Figure 32. eSales - Display Error Process

When the eSales - Display Error Process is called, the following events happen:

- The process sets the error message.
- It displays the error message in the General Error Message view.
- It returns to the original workflow.



## eSales - Login View

The process for displaying the Login view is shown in [Figure 33](#).

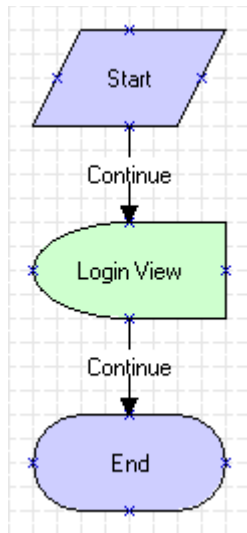


Figure 33. eSales - Login View

This process is called by the eSales - Checkout Process (see [Figure 8 on page 76](#)) when an anonymous user tries to check out. The eSales - Login View process is necessary because the Login view is based on the Login business object, while the eSales - Checkout Process is based on the Catalog business object.

## eSales - Order Confirmation View

The process for displaying the Order Confirmation view is shown in [Figure 34](#).

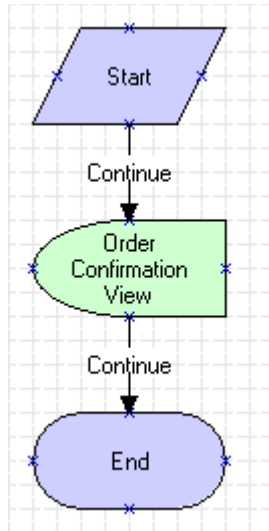


Figure 34. eSales - Order Confirmation View

This process is called by the eSales - Authorize Order Process (see [Figure 15 on page 87](#)). The eSales - Order Confirmation View process is necessary because the Order Confirmation view is based on the Order Entry (Sales) business object, while the eSales - Authorize Order Process is based on the Catalog business object.

## SS Order Confirmation View Process

The workflow for displaying the Order Confirmation view is shown in [Figure 35](#).

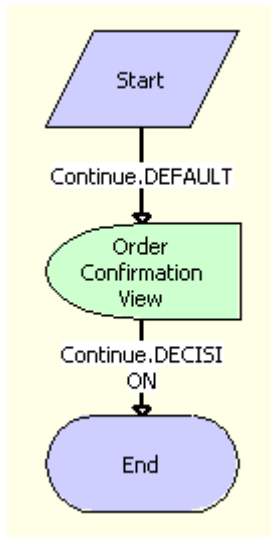


Figure 35. SS - Order Confirmation View

This process is called by the SS - Authorize Order Process (see [Figure 16 on page 89](#)). The eSales - Order Confirmation View process navigates the user to the Order Confirmation View.

## eSales - Shopping Cart View

The process for displaying the Shopping Cart is shown in [Figure 36](#).

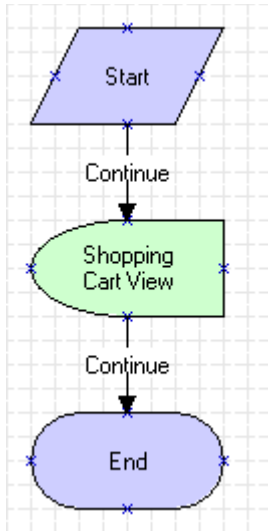


Figure 36. eSales - Shopping Cart View

This process is called by the eSales - Modify Order Process (see [Figure 30 on page 101](#)). The eSales - Shopping Cart View process is necessary because the Shopping Cart is based on the Catalog business object, while the eSales - Modify Order Process is based on the Order Entry (Sales) business object.

## Auction Workflow

Auctions are not part of the current Siebel Business Applications functionality. An auction workflow is included for backward compatibility only.

Quotes of type Auction Quote (that is, winning bids) are converted to orders automatically after a given period of time. This is done using a workflow policy such as Complete Auction Order, which converts the quote to an order after two days by triggering the Silent Authorize Order workflow policy action.

For more information on auction quotes, see [“Quote Types in Siebel eSales” on page 57](#). For information on workflow policies, see *Siebel Business Process Designer Administration Guide*.

## eSales - Silent Authorize Order Process

The eSales - Silent Authorize Order Process, shown in [Figure 37](#), is triggered by the Silent Authorize Order workflow policy action.

**NOTE:** Auctions are not part of the current Siebel Business Applications functionality. This workflow is included for backward compatibility only.

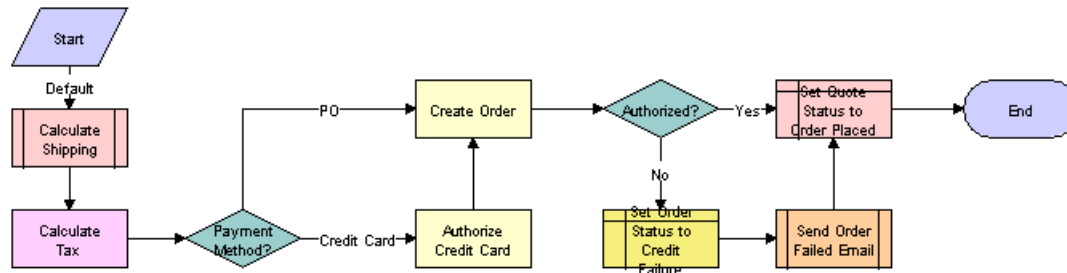


Figure 37. eSales - Silent Authorize Order Process

When this workflow is triggered, the following events happen:

- The Shipping Calculation Process is called. For more information, see [Chapter 3, "Setting Up Siebel eSales."](#)
- The Calculate Tax business service is called.
- The process determines the payment method. If a credit card is used, the card is authorized.
- An order is created.
- If credit card authorization fails, the order status is set to Credit Failure and an email message is sent to the customer indicating the problem.
- The quote status is set to Order Placed.



# A

## Shopping Service Methods Reference

This appendix describes the Shopping Service business service, and it contains the following sections:

- [“About the Shopping Service Business Service” on page 111](#)
- [“Siebel eSales User Action Methods” on page 111](#)
- [“Siebel eSales Workflow Methods” on page 128](#)

### About the Shopping Service Business Service

In Siebel eSales, the Shopping Cart and order management use workflows for various actions, such as checking out, editing quotes, modifying orders, and so on.

The Shopping Service business service provides methods that can be invoked in the workflows. In addition, in Siebel 7 the Add to Cart feature has been moved from the Quote business component to the Shopping Service business service. This is because business services can be accessed more universally than business components.

Therefore, the Shopping Service can support other features such as Add to Favorites, Quick Add, and Save Cart As Template. It is also easier for other applications, such as Siebel Partner Relationship Management, to integrate with Siebel eSales order management.

### Siebel eSales User Action Methods

These methods are invoked by actions taken by users of the Siebel eSales Web site, such as clicking Add to Cart. They are as follows:

- [“AddItem Method” on page 112](#)
- [“AddToCart Method” on page 112](#)
- [“AddToFavoriteList Method” on page 120](#)
- [“SetShoppingCart Method” on page 122](#)
- [“SetApplicationName Method” on page 124](#)
- [“OrderTemplate Method” on page 126](#)
- [“SetShoppingCartInfoChanged Method” on page 125](#)
- [“GetCartInfo Method” on page 125](#)

## AddItem Method

This method is similar to [“AddToCart Method” on page 112](#).

The only difference is that AddItem can be used to AddItems to the shopping cart or to favorites, while AddToCart can only be used to add items to the shopping cart

## AddToCart Method

This method is invoked to add an item to the Shopping Cart. It checks the input argument Action. Depending on the value of Action, the Shopping Cart takes different actions, as described in [Table 15](#).

Table 15. Shopping Cart Actions

| Action              | Method Invoked   |
|---------------------|------------------|
| AddToCart           | DefaultAddToCart |
| Customize           | DefaultAddToCart |
| QuickAdd            | QuickAdd         |
| AddTrainingItem     | AddTrainingItem  |
| AddEventItem        | AddTrainingItem  |
| AddEventSessionItem | AddTrainingItem  |

What the AddToCart method does is to create one or more records (for complex products) in the Quote Item business component in the current quote record (indicated by CurrentQuoteId) in the Quote business component.

If the product has extended attributes, it also copies the extended attribute values from the Product XA VBC business component to the Quote Item XA business component. The extended attributes for a product are in the XA Attribute Value business component. End users can select values for each modifiable attribute; these selections are stored in the Product XA VBC business component.

For example, if you add a product with four attributes to the Shopping Cart, you create one record in the Quote Item business component and four records in the Quote Item XA business component. The mapping is as follows:

- Product to Quote Item
- Product XA VBC to Quote Item XA



## Preconditions for Calling the AddToCart Method

- **Where AddToCart can be called.** This method can be called from either an applet or a business component. In each case, the caller function needs to call the SetBusComp method to set the current business component to the Shopping Service instance.

The Shopping Service instance needs to use the current business component to get the current business object and all the context information. Therefore, you cannot invoke this method from an applet that does not have a business component.

- **Quote record.** The Shopping Cart is just a quote record whose row ID equals the CurrentQuoteId profile attribute. The Quote Type field value for this current quote record can be Quotation, Persistent Cart, or Anonymous Quote.
  - If it is Quotation, then the current quote is a normal quote.
  - If it is Persistent Cart, it means this application supports the Persistent Shopping Cart and this quote is the Persistent Shopping Cart for that user.
  - If it is Anonymous Quote, it means this application allows anonymous users to add items to the Shopping Cart and this user is an anonymous user.
- **Price List Id, Promotion Id, and Campaign Id fields.** The current quote record has a Price List Id field, a Promotion Id field, and a Campaign Id field. The field value for Price List Id cannot be empty. The field values for Promotion Id and Campaign Id can be empty or not.

There are also three corresponding profile attributes—PromotionId, CampaignId, and PriceListId. These can be set by Siebel eMarketing when a user clicks the hyperlink on a Web offer. Therefore, the field values for Price List Id, Campaign Id, and Promotion Id in the current quote record can be different from the values in the PriceListId, CampaignId, and PromotionId profile attributes. When users add items to the Shopping Cart, these three field values in the current profile quote record are set to the value from the corresponding profile attributes.

- **Internal Product by Price List Optional business component.** This is the underlying product business component for AddToCart, QuickAdd, and Customize.

All other business components use this business component as the proxy business component for AddToCart. This is because only this business component has extended attribute child business components such as Product XA VBC and XA Attribute Value. In addition, it is used in Attribute Adjustments.

The user properties for the Internal Product by Price List Optional business component are listed in [Table 16](#).

Table 16. Internal Product by Price List Optional BC User Properties

| Name                    | Value             | Explanation   |
|-------------------------|-------------------|---|
| Field:Base Price        | Catalog Price     | The C++ code for the Shopping Service and Attribute Adjustments refers to certain fields. Instead of hard-coding the field name in C++, this user property is used for mapping between the field to which the C++ code refers and the actual field name in the business component.<br><br>For example, this user property indicates that the Base Price to which the C++ code refers is called Catalog Price in the Internal Product by the Price List Optional business component.<br><br>This way, Base Price in the C++ code can be mapped to different fields in different business components. |
| Field:Catalog Price     | Catalog Price     | Similar to Field:Base Price. Used in Attribute Adjustments.   |
| Field:Reference Price   | Reference Price   | Similar to Field:Base Price. Used in Attribute Adjustments.   |
| Field:Price Book Id     | Price Book Id     | Similar to Field:Base Price. Used in Attribute Adjustments.   |
| Field:Promotional Price | Promotional Price | Similar to Field:Base Price. Used in Attribute Adjustments.   |
| Field:Quantity          | Quantity          | Similar to Field:Base Price. Refers to the Quantity field. Used in Attribute Adjustments and in the Shopping Service.   |
| Field:Your Price        | Your Price        | Similar to Field:Base Price. Used in Attribute Adjustments.   |
| Product XA              | Product XA VBC    | Indicates that the extended attribute to use is Product XA VBC.   |

Table 16. Internal Product by Price List Optional BC User Properties

| Name                  | Value | Explanation   |
|-----------------------|-------|---|
| SetPriceListSearch    | Y     | Indicates that the [Price List Id] = GetProfileAttr("PriceListId") search specification should be applied to the business component.<br><br>If used, only products that are on the current price list are visible in Siebel eSales. |
| Use Promotional Price | Y     | Used in Attribute Adjustments.<br><br>When this is Y, the Base Price equals the Promotional Price if the Promotional Price field is not empty. Otherwise, the Base Price equals the Reference Price.                                |

- **AddToCartAutoQuote input argument.** When there is no current quote as specified by CurrentQuoteId, AddToCart decides whether to create a new quote automatically based on certain conditions:
  - AddToCart checks the AddToCartAutoQuote input argument. If its value is Y or TRUE, AddToCart automatically creates a new quote. If its value is N or FALSE, AddToCart does not automatically create a new quote.
  - If the AddToCartAutoQuote input argument is empty, AddToCart checks the AddToCartAutoQuote server component parameter ([Table 17 on page 116](#)).
- **AddToCartGotoView input argument.** The calling action can also specify the next view to which to navigate after finishing the AddToCart method. This is done through the AddToCartGotoView input argument:
  - If this input is not empty, the application navigates to the specified view.
  - If this input is empty, the Shopping Service looks for the AddToCartGotoView server component parameter as defined for InfraShoppingService in the repository ([Table 17](#)).

- **Server component parameters.** The server component parameters used by the Shopping Service business service are listed in [Table 17](#). You can change these parameters by navigating to Server Administration > Components > Component Parameters.

Table 17. Server Component Parameters in the Shopping Service

| Name                   | Default Value                           | Explanation   |
|------------------------|---|---|
| AddToCartAutoQuote     | TRUE                                    | The value can be TRUE or FALSE.<br><br>If it is TRUE, AddToCart automatically creates a new quote if there is no current quote.   |
| AddToCartGotoView      | NONE                                    | The value can be NONE or a specific view name.<br><br>If it is a specific view name, the application navigates the user to that view after finishing the AddToCart method.  |
| AnonymousQuote         | TRUE                                    | The value can be TRUE or FALSE.<br><br>If it is TRUE, AddToCart allows anonymous users to add items to the Shopping Cart, and it creates quotes with quote type Anonymous Quote.<br><br>If it is FALSE, AddToCart does not allow anonymous users to add items to the Shopping Cart.   |
| AutoQuoteDefaultOwner  | TRUE                                    | The value can be TRUE or FALSE.<br><br>If it is TRUE, when AddToCart automatically creates a new quote, it automatically fills in the Contact Id and Account Id fields with the current user's login ID and account ID.   |
| PersistentShoppingCart | FALSE                                   | The value can be TRUE or FALSE.<br><br>If it is TRUE, the application supports the Persistent Shopping Cart. The quote created by AddToCart will have type Persistent Cart.<br><br>When the user logs in to the application, the runtime event triggers the eSales Web Login Action Set. The SetShoppingCart method in that action set loads the quote with type Persistent Cart for that user. |
| ProductDetailView      | Product Detail - Features View (eSales) | Specifies the view name for the Product Detail view.  |
| ShoppingCartView       | NONE                                    | Specifies the view for the Shopping Cart.   |

- **Business object must include all necessary business components.** The list of business components that must be in the business object calling the AddToCart method is shown in [Table 18](#), using the Catalog business object as an example.

Table 18. Necessary Business Components in the Business Object

| Business Component                        | Link  | Comment   |
|---|---|---|
| Internal Product by Price List Optional   |   | Underlying business component for AddToCart, QuickAdd, and Customize.<br><br>All other business components by default use this business component as the AddToCart Proxy Buscomp. |
| Internal Product by Price List Optional 2 | Catalog Category/Internal Product by Price List Optional 2  | Used in the Product list applet for AddToCart.<br><br>This business component uses Internal Product by Price List Optional as the AddToCart Proxy Buscomp.                        |
| XA Attribute Value                        | Internal Product by Price List Optional/XA Attribute Value  | Needed for extended attributes.   |
| Product XA VBC                            |   | Needed for extended attributes.   |
| Product XA VBC Form                       | Internal Product by Price List Optional/Product XA VBC Form | Needed for extended attributes.   |
| Quote                                     |   |   |
| Quote Item                                | Quote/Quote Item  |   |
| Quote Item XA                             |   |   |
| Training Class Registration               |   | Needed for AddTrainingItem.   |
| eEvents Event Attendee                    |   | Needed for AddTrainingItem.   |
| eEvents Session Attendee                  |   | Needed for AddTrainingItem.   |
| Favorite List                             |   | Needed for AddToFavoriteList.   |
| Favorite List Item                        | Favorite List/Favorite List Item                            | Needed for AddToFavoriteList.   |

## Arguments for the AddToCart Method

The AddToCart method has the arguments shown in [Table 19](#).

Table 19. AddToCart Arguments

| Name                   | Type   | Data Type | Optional | Comments  |
|------------------------|--------|-----------|----------|---|
| Action                 | Input  | String    | Y        | Specifies the action to take.<br><br>Action can take the following values: <ul style="list-style-type: none"> <li>■ AddToCart</li> <li>■ Customize</li> <li>■ QuickAdd</li> <li>■ AddTrainingItem</li> <li>■ AddEventItem</li> <li>■ AddEventSessionItem</li> </ul> |
| AddToCart Quote Fields | Input  | String    | Y        | Concatenated string of Quote business component field name/value pairs.<br><br>AddToCart deconstructs this string and sets the corresponding fields in the Quote business component.  |
| AddToCart AutoQuote    | Input  | String    | Y        | Used to decide whether to create a new quote automatically if there is no current quote.<br><br>If empty, AddToCart looks for the server component parameter with the same name.  |
| AddToCart GotoView     | Input  | String    | Y        | Used to decide to which view to navigate after adding an item has been completed.<br><br>If empty, AddToCart looks for the server component parameter with the same name.   |
| Source Currency Code   | Input  | String    | Y        | Specifies the currency code for the price.<br><br>If not empty and not equal to the currency code in the quote, AddToCart does the currency conversion and sets the converted Price field in the quote item.  |
| Quote Id               | Output | String    | Y        | Returns the quote ID for the current quote.   |
| Quote Item Id          | Output | String    | Y        | Returns the quote item ID.  |
| ReturnVal              | Output | String    | Y        | Returns Success or Fail. Not used.  |

## Pseudo-Algorithm for AddToCart

Follow these steps when using a pseudo-algorithm for AddToCart:

- 1 Call CanAddToCart method to check whether or not a user can add items to the Shopping Cart:  
If the CurrentQuoteId profile attribute is not empty and is a valid quote ID, return TRUE;  
else if the AddToCartAutoQuote input argument is TRUE, return TRUE;  
else if the AddToCartAutoQuote server component parameter is TRUE, return TRUE;  
else return FALSE.
- 2 Check the Action input argument:  
If Action = AddTrainingItem or AddEventItem or AddEventSessionItem, call the AddTrainingItem method;  
Else if Action = QuickAdd, call the QuickAdd method;  
Else if Action = AddToCart or Customize:  
    Call the DefaultAddToCart method.
- 3 Call the PostAddToCart method.

## Pseudo-Algorithm for DefaultAddToCart

Follow these steps when using a pseudo-algorithm for DefaultAddToCart:

- 1 Get the Quote, Quote Item, and Quote Item XA business components.
- 2 Call AddHeader to make sure there is a current quote for adding an item, What AddHeader does is to make sure there is a current quote, if there is not a current one. AddHeader will create a new quote and set the CurrentQuoteID profile attribute value to the Row ID of the newly created quote record.
- 3 Create a new record in the Quote Item business component.
- 4 Populate that line item with information from Product the user has selected.
- 5 Check the Action input argument:  
If Action = Customize, then call ReConfigCxProd to launch the customizable product run-time view.
- 6 End.

## Pseudo-Algorithm for PostAddToCart

Follow these steps when using a pseudo-algorithm for PostAddToCart:

- 1 Check the Action input argument:  
If Action = Customize, End;  
Else go to step 2.

- 2 Check the AddToCartGotoView input argument:  
If not empty, then navigate to that view and End;  
Else go to step 3.
- 3 Check the AddToCartGotoView server component parameter:  
If not empty and <> NONE, then navigate to that view and End;  
Else End.

## AddToFavoriteList Method

This method is invoked to add items to the Favorites list.

### Preconditions for Calling the AddToFavoriteList Method

When calling this method, you need to make sure the business components that are shown in [Table 20](#) are in the business object, using the Catalog business object as an example.

Table 20. Necessary Business Components in the Business Object

| Business Component                        | Link   | Comment   |
|---|--|---|
| Internal Product by Price List Optional   |  | Product business component used in Siebel eSales. |
| Internal Product by Price List Optional 2 | Catalog Category/Internal Product by Price List Optional 2 | Business component used in product list applet.   |
| Favorite List                             |  | Favorite List business component.                 |
| Favorite List Item                        | Favorite List/Favorite List Item                           | Favorite List Item business component.            |



## Arguments for the AddToFavoritesList Method

The AddToFavoritesList method has the arguments shown in [Table 21](#).

Table 21. AddToFavoritesList Arguments

| Name       | Type   | Data Type | Optional | Comments  |
|------------|--------|-----------|----------|---|
| Action     | Input  | String    | Y        | Equals AddToFavoritesList.  |
| Product Id | Input  | String    | Y        | If this input is not empty, add the product specified by this ID to the Favorites list.<br><br>Otherwise, assume the current record in the current business component is the item to add to the Favorites list. |
| GotoView   | Input  | String    | Y        | If this input is not empty, AddToFavoritesList navigates to this view after completing the AddToFavoritesList action.   |
| RetVal     | Output | String    | Y        | Returns Success or Fail. Not used.  |

### Pseudo-Algorithm for AddToFavoritesList

- 1 If the user is an anonymous user, End.
- 2 Get the Favorite List and Favorite List Item business components.
- 3 Query the Favorite List business component with the search specification [Person Id] = Logi nId;  
If there is a existing record, go to step 4;  
Else create a new record and set [Person Id] to current login ID.
- 4 Query the Favorite List Item business component with the search specification [Product Id] = i nput product ID;  
If there is an existing record, go to step 5;  
Else create a new record and set [Product Id] to input product ID.
- 5 If GotoView input argument is not empty, navigate to the specified view and End;  
Else check the Favorite List View user property in the Shopping Service business service:  
If this value is empty, End;  
Else navigate to this view and End.

## SetShoppingCart Method

This method is used in run-time event action sets to support the Persistent Shopping Cart and anonymous users adding items to the Shopping Cart.

SetShoppingCart is used in the eSales WebLogin Action Set. This action set is associated with the WebLogin run-time event for the Siebel eSales application object. When a Siebel eSales user logs in, the WebLogin run-time event is triggered. This in turn triggers the eSales Web Login Action Set and executes the SetShoppingCart method.

The method checks if there is already a current quote in the session as indicated by the CurrentQuoteId profile attribute:

- If this profile attribute is empty, it determines if there is any quote of type Persistent Cart associated with the user. If there is one, it loads that quote and sets the CurrentQuoteId profile attribute to that quote ID.
- If CurrentQuoteId is already set, it checks if the quote type is Anonymous Quote and converts it to Persistent Cart if it is.

### Preconditions for Calling the SetShoppingCart Method

- The following server component parameters must be set to TRUE:
  - PersistentShoppingCart
  - AnonymousQuote
- To use this method, you need to create an action set with an action that invokes this method. For example, the actions in the eSales Web Login Action Set are shown in [Table 22](#).

Table 22. eSales Web Login Action Set Actions

| Action          | Business Service | InvokeMethod    |
|-----------------|------------------|-----------------|
| SetPriceList    | Pricing Manager  | SetPriceList    |
| SetShoppingCart | Shopping Service | SetShoppingCart |

After that, you need to associate this action set with the WebLogin run-time event for the application. The following applications use the eSales WebLogin Action Set:

- Siebel Partner Relationship Management
- Siebel eCustomer
- Siebel eSales

### Arguments for the SetShoppingCart Method

None.

## Pseudo-Algorithm for SetShoppingCart

- 1 If the Is Anonymous profile attribute = TRUE, End.
- 2 Check the CurrentQuoteId profile attribute:  
If empty or = EMPTY\_CART, there is no current quote. Go to step 3;  
Else go to step 4.
- 3 Get the Catalog business object and the Quote, Quote Item, and Quote Item XA business components. Query the Quote business component with the search specification [Contact Id] = Logi nI d ( ) AND [Quote Type] = LookupVal ue ( ' QUOTE\_TYPE', ' Persi stent Cart' ) to find the quote record of type Persistent Cart for that user. One or zero records will be returned.  
  
If one record is returned, the user has a Persistent Cart. Set the CurrentQuoteId profile attribute to the record ID, then go to step 8.
- 4 Get the Catalog business object and the Quote, Quote Item, and Quote Item XA business components. Query the Quote business component with the search specification [Id] = GetProfi leAttr("CurrentQuoteId") to get the current quote. Check the Quote Type field value:  
  
If [Quote Type]= LookupValue ( 'QUOTE\_TYPE', 'Anonymous Quote'), the current quote is an anonymous user's Shopping Cart. Go to step 5;  
  
Else, go to step 6.
- 5 Before converting Anonymous Quote to Persistent Cart, it is necessary to check if the user already has a Persistent Shopping Cart. Get a second instance of the Catalog business object and the Quote, Quote Item, and Quote Item XA business components. Query the second instance of the Quote business component with the search specification [Contact Id] = Logi nI d ( ) AND [Quote Type] = LookupVal ue ( ' QUOTE\_TYPE', ' Persi stent Cart' ). It will return one or zero records.  
  
If it returns no records, go to step 6;  
  
If it returns one record, it is necessary to resolve the conflict between the anonymous quote and an existing persistent Shopping Cart. Go to step 7.
- 6 Convert Anonymous Quote to Persistent Cart:  
Set [Account Id] to the account ID of the current user;  
Set [Contact Id] to the login ID of the current user;  
Set [Quote Type] to LookupValue ( 'QUOTE\_TYPE', 'Persistent Cart' );  
Set [Price List Id] to the current price list ID;  
End.
- 7 There are two quote records: the quote record in the first Quote instance is the Anonymous Quote; the record in the second Quote instance is the previous Persistent Shopping Cart. Save the previous Persistent Shopping Cart as a quote with a name.
- 8 If the price list in the current quote is different from the current price list, set [Price List Id] to the current price list ID in the system.

## SetApplicationName Method

This method sets the ApplicationName profile attribute, which is used in workflow branching conditions and in Siebel Personalization to hide or show an applet in a view for various applications.

SetApplicationName is used in the eSales WebSessionStart Action Set. This action set is associated with the WebSessionStart run-time event for the Siebel eSales application object. When a Web session starts (usually the first time a user visits the Web site), this event is triggered. This in turn triggers the eSales Web Session Start Action Set, executing the SetApplicationName method. The ApplicationName profile attribute is set to the name for the application that is running.

### Preconditions for Calling the SetApplicationName Method

To use this method, you need to create an action set and invoke this method from an action in the action set, as shown for the eSales WebSessionStart Action Set in [Table 23](#).

Table 23. Sales WebSessionStart Action Set

| Action             | Business Service | InvokeMethod       |
|--------------------|------------------|--------------------|
| SetApplicationName | Shopping Service | SetApplicationName |

The following applications use the eSales WebSessionStart Action Set:

- Siebel Call Center
- Siebel Partner Relationship Management
- Siebel eCustomer
- Siebel eEvents Management
- Siebel eSales
- Siebel Self Service
- Siebel Marketing

### Arguments for the SetApplicationName Method

None.

### Pseudo-Algorithm for SetApplicationName

Get the application name and set the ApplicationName profile attribute.

## SetShoppingCartInfoChanged Method

This method monitors any change in the number of items in the shopping cart. If an item in the shopping cart is added or deleted, then this method assigns the value Y to the ShoppingCartInfoChanged profile attribute, regardless of the previous value.

### Preconditions for Calling the SetShoppingCartInfoChanged Method

This method is called while adding line items to, or, deleting line items from the shopping cart, so that the current value of the item count in the cart is shown.

### Arguments for the SetShoppingCartInfoChanged Method

There are no arguments for the SetShoppingCartInfoChanged Method.

### Pseudo-Algorithm for SetShoppingCartInfoChanged

Assigns the ShoppingCartInfoChanged profile attribute to Y.

## GetCartInfo Method

This method retrieves the number of root products present in the cart, and is called when items are added to or deleted from the cart.

### Preconditions for Calling the GetCartInfo Method

This method is invoked to obtain the number of current items in the cart. The active quote must exist and the CurrentQuoteId profile attribute must be populated, otherwise the number of current items in the cart returns as 0. The Quote Item business component and the Quote business object are passed to this method as an optional input argument. In the event that these items are not passed to the method, then by default the GetCartInfo method takes Quote as a business object and SS Quote Item as a business component.

### Arguments for the GetCartInfo Method

The GetCartInfo method has the arguments shown in [Table 24](#).

Table 24. GetCartInfo Arguments

| Name  | Type   | Data Type | Optional | Comments   |
|-------|--------|-----------|----------|--|
| Count | Output | String    | N        | Returns count of root items present in the current cart. |

Table 24. GetCartInfo Arguments

| Name      | Type  | Data Type | Optional | Comments   |
|-----------|-------|-----------|----------|--|
| SSBusComp | Input | String    | Y        | Specifies the business component to be queried.                                |
| SSBusObj  | Input | String    | Y        | Specifies the business object containing the business component to be queried. |

### Pseudo-Algorithm for GetCartInfo

- If the SSBusComp argument is not passed, this method reads the value of CurrentQuoteId profile attribute to obtain the current quote ID and to query the SS Quote Item business component. With the additional search specification, this retrieves the number of root line items present in the quote.
- Sets the record count as output of the method. If the additional search specification has a value defined as part of the Cart Count Search Spec business service user property, then this value overwrites the Root Quote Items search specification, and is then used, along with Quote Id, to get the line item for the current quote.

## OrderTemplate Method

OrderTemplate copies all the line items from a template to the Shopping Cart. The behavior is similar to AddToCart; the difference is that when you order a template, you add several line items to the Shopping Cart. Each line item will have the quantity specified in the template.

A template is a quote of type Template, so template records are a subset of quote records. Templates have a similar structure to quotes—there is a 1:1 mapping between template and quote business components:

- Quote Template to Quote
- Quote Template Item to Quote Item
- Quote Template Item XA to Quote Item XA

## Preconditions for Calling the OrderTemplate Method

OrderTemplate uses field maps defined in the Shopping Service business service user properties to copy field values from Quote Template Item to Quote Item and from Quote Template Item XA to Quote Item XA. The field maps are shown in [Table 25](#) and [Table 26](#).

Table 25. Quote Template Item to Quote Item Field Map

| Name  | Value   |
|---|---|
| Quote Template Item Quote Item FieldMap 100 | Product Id Product Id                             |
| Quote Template Item Quote Item FieldMap 101 | Quantity Requested Quantity Requested             |
| Quote Template Item Quote Item FieldMap 102 | Adjusted Reference Price Adjusted Reference Price |
| Quote Template Item Quote Item FieldMap 103 | Base Price Base Price                             |
| Quote Template Item Quote Item FieldMap 104 | Unit Price Unit Price                             |
| Quote Template Item Quote Item FieldMap 105 | Product Product                                   |
| Quote Template Item Quote Item FieldMap 106 | Product Port Id Product Port Id                   |
| Quote Template Item Quote Item FieldMap 107 | End   |

Table 26. Quote Template Item XA to Quote Item XA Field Map

| Name  | Value                     |
|---|---------------------------|
| Quote Template Item XA Quote Item XA FieldMap 100 | Attribute Id Attribute Id |
| Quote Template Item XA Quote Item XA FieldMap 101 | Name Name                 |
| Quote Template Item XA Quote Item XA FieldMap 102 | Display Name Display Name |
| Quote Template Item XA Quote Item XA FieldMap 103 | Read Only Read Only       |
| Quote Template Item XA Quote Item XA FieldMap 104 | Data Type Data Type       |
| Quote Template Item XA Quote Item XA FieldMap 105 | LOV Type LOV Type         |
| Quote Template Item XA Quote Item XA FieldMap 106 | Value Value               |
| Quote Template Item XA Quote Item XA FieldMap 107 | End                       |

## Arguments for the OrderTemplate Method

The OrderTemplate method has the arguments shown in [Table 27](#).

Table 27. OrderTemplate Arguments

| Name        | Type   | Data Type | Optional | Comments  |
|-------------|--------|-----------|----------|---|
| Template Id | Input  | String    | Y        | Not used, as OrderTemplate is always row-sensitive.                   |
| ReturnVal   | Output | String    | Y        | Returns Success or Fail depending on whether AddItem succeeds or not. |

## Pseudo-Algorithm for OrderTemplate

- 1 Get the Quote Template, Quote Template Item, Quote Template Item XA, Quote, Quote Item, and Quote Item XA business components.

The Quote Template business component is already positioned on the correct record, as the OrderTemplate method is row-sensitive.

- 2 Call the AddHeader method to make sure there is a Shopping Cart.
- 3 Loop through each record in the Quote Template Item business component and create a corresponding record in the Quote Item business component. Use the field maps to copy fields from Quote Template Item to Quote Item, then copy the child records from Quote Template Item XA to Quote Item XA.

**NOTE:** As Quote Item and Quote Template Item have hierarchies, you cannot simply loop through the records in Quote Template Item and copy them to Quote Item. You need to maintain the hierarchical relationship. To do this, you need to copy the parent record before you copy its child records.

In the Shopping Service implementation, records are first loaded from Quote Template Item and a tree in the memory is generated. The tree is then traversed from the root, so the parent record is copied before the child records.

## Siebel eSales Workflow Methods

The following methods are invoked by Oracle's Siebel eSales workflows:

- "SaveCartAsTemplate Method" on page 129
- "SaveCartAsQuote Method" on page 131
- "DiscardCart Method" on page 133
- "OrderQuote Method" on page 135
- "ValidQuote Method" on page 136
- "RemoveInvalidItems Method" on page 137



- [“CreateOrder Method” on page 138](#)
- [“Reorder Method” on page 140](#)
- [“GetShoppingCartStatus Method” on page 142](#)
- [“MoveToCart Method” on page 143](#)
- [“RepriceQuote Method” on page 144](#)
- [“SelectPrimary Method” on page 145](#)
- [“SetBusCompFields Method” on page 146](#)
- [“ValidatePayment Method” on page 147](#)
- [“ValidateAddress Method” on page 148](#)
- [“SetPendingRequest Method” on page 148](#)

**NOTE:** Some of the methods in this section have values pertaining to auctions. Auctions are not part of the current Siebel Business Application’s functionality. These values are included for backward compatibility only.

## SaveCartAsTemplate Method

SaveCartAsTemplate generates a template from a quote. The implementation is Template (see [“OrderTemplate Method” on page 126](#)) with some special considerations—for example, you cannot copy a Siebel Training or Siebel Events Management line item into a template.

### Preconditions for Calling the SaveCartAsTemplate Method

Field maps are used to copy fields from the Quote to Quote Template, Quote Item to Quote Template Item, and Quote Item XA to Quote Template Item XA business components.

Sample field maps are shown in [Table 28](#). See the Shopping Service business service user properties in Siebel Tools for more details.

Table 28. Sample Field Maps for the SaveCartAsTemplate Method

| Name  | Value                     |
|---|---------------------------|
| Quote Quote Template FieldMap 100                 | Contact Id Contact Id     |
| Quote Item Quote Template Item FieldMap 100       | Product Id Product Id     |
| Quote Template Item XA Quote Item XA FieldMap 100 | Attribute Id Attribute Id |

## Arguments for the SaveCartAsTemplate Method

The SaveCartAsTemplate method has the arguments shown in [Table 29](#).

Table 29. SaveCartAsTemplate Arguments

| Name              | Type   | Data Type | Optional | Comments  |
|-------------------|--------|-----------|----------|---|
| Quote Name        | Input  | String    | Y        | Saves the Shopping Cart as a template with this name.   |
| Quote Name Field  | Input  | String    | Y        | Tells the Shopping Service which field is the quote name.<br><br>By default, it is Name if this input argument is not used.   |
| Return Error Code | Input  | String    | Y        | If equal to N, the method will not return errors.<br><br>When a method returns an error, the workflow stops. To avoid stopping the workflow, you can set this input to N. This way, the Shopping Service will not return an error but instead save the error message in the output argument Error Message. You can then handle the error message as you want. |
| Error Message     | Output | String    | Y        | Stores error messages.  |
| ReturnVal         | Output | String    | Y        | Returns Success or Fail. Not used.  |

## Pseudo-Algorithm for SaveCartAsTemplate

- 1 Get the Quote Template, Quote Template Item, Quote Template Item XA, Quote, Quote Item, and Quote Item XA business components.

The Quote Template business component is already positioned on the correct record, as the OrderTemplate method is row-sensitive.

- 2 Check the Quote Name input argument. Query the Quote business component to see if there is an existing record with the same name.

If there is, go to step 3;

Else go to step 4.

As Quote Template is a special type of Quote, the Quote business component can be used to search both Quote and Quote Template.

- 3 Set Error Message to indicate there is a duplicate name. Check the Return Error Code input argument.  
If Return Error Code = N, do not return an error;  
Else return an error.
- 4 Create a record in the Quote Template business component. Use the Quote to Quote Template field maps to copy fields from the record in the Quote business component to the record in the Quote Template business component. Set the Name to the Quote Name input argument.
- 5 Copy each record in the Quote Item business component to the Quote Template Item business component.

**NOTE:** As in the OrderTemplate method, make sure the hierarchical relationship is copied correctly by copying each parent record before its child records. Also, copy the child records from Quote Item XA to Quote Template XA.

## SaveCartAsQuote Method

SaveCartAsQuote saves the Shopping Cart (current quote record) as a quote.

### Preconditions for Calling the SaveCartAsQuote Method

- This method is invoked in the following workflows:
  - eSales - Save Cart Process
  - eSales - Edit Quote Process
  - eSales - Order Quote Process
  - eSales - Modify Order Process
  - eSales - Reorder Process
- The field value of Quote Type for the current quote record can be Persistent Cart, Auction Quote, Quotation, or Anonymous Quote. Anonymous users cannot call SaveCartAsQuote, so only the first three quote types can be saved.

If the quote type is Persistent Cart, SaveCartAsQuote changes the type to Quotation. Otherwise, it does not change the quote type.

## Arguments for the SaveCartAsQuote Method

The SaveCartAsQuote method has the arguments shown in [Table 30](#).

Table 30. SaveCartAsQuote Arguments

| Name              | Type   | Data Type | Optional | Comments  |
|-------------------|--------|-----------|----------|---|
| Quote Name        | Input  | String    | Y        | Saves the Shopping Cart as a quote with this name.  |
| Quote Name Field  | Input  | String    | Y        | Tells the Shopping Service which field is the quote name.<br><br>By default, it is Name if this input argument is not used.   |
| Return Error Code | Input  | String    | Y        | If equal to N, the method will not return errors.<br><br>When a method returns an error, the workflow stops. To avoid stopping the workflow, you can set this input to N. This way the Shopping Service will not return an error, but instead save the error message in the output argument Error Message. You can then handle the error message as you want. |
| Error Message     | Output | String    | Y        | Stores error messages.  |
| ReturnVal         | Output | String    | Y        | Returns Success or Fail. Not used.  |

## Pseudo-Algorithm for SaveCartAsQuote

- 1 Get the Quote business component.
- 2 Get the Id of the current record in the Quote business component:
  - If Id <> CurrentQuoteId profile attribute, go to step 3;
  - Else go to step 4.
- 3 SaveCartAsQuote is being invoked from the eSales - Edit Quote Process, eSales - Order Quote Process, eSales - Modify Order Process, or eSales - Reorder Process workflow. The current record in the Quote business component is the quote being acted upon, and it is not the current quote in the Shopping Cart.
 

Get another instance of the Catalog business object and the Quote business component, and position the second instance of the Quote business component on the quote record as indicated by the CurrentQuoteId profile attribute. As the first instance of the Quote business component is not being acted upon, from now on Quote means the second instance of the Quote business component.

- 4 Check the Quote Name input argument. Query the Quote business component to see if there is an existing record with the same name:  
If there is, go to step 5;  
Else go to step 6.
- 5 Set the Error Message output argument to indicate that there is a duplicate name. Check the Return Error Code input argument:  
If Return Error Code = N, do not return an error;  
Else return an error.
- 6 Set the Name field with the Quote Name input argument. Check the field value of [Quote Type]:  
If it is Persistent Cart, then change it to Quotation.

## DiscardCart Method

DiscardCart discards the Current Quote and clears the CurrentQuoteId profile attribute. If the current quote type is Persistent Cart or Anonymous Quote, it also deletes the record from the Quote business component.

### Preconditions for Calling the DiscardCart Method

- This method is invoked in the following workflows:
  - eSales - Edit Quote Process
  - eSales - Order Quote Process
  - eSales - Modify Order Process
  - eSales - Reorder Process

It is also invoked by the Empty Cart button in the Shopping Cart.

- The field value of Quote Type for the current quote record can be Persistent Cart, Auction Quote, Quotation, or Anonymous Quote. When the quote type is Persistent Cart or Anonymous Quote, the quote record is deleted.

## Arguments for the DiscardCart Method

The DiscardCart method has the arguments shown in [Table 31](#).

Table 31. DiscardCart Arguments

| Name              | Type   | Data Type | Optional | Comments  |
|-------------------|--------|-----------|----------|---|
| Return Error Code | Input  | String    | Y        | If equal to N, the method will not return errors.<br><br>When a method returns an error, the workflow stops. To avoid stopping the workflow, you can set this input to N. This way the Shopping Service will not return an error, but instead save the error message in the output argument Error Message. You can then handle the error message as you want. |
| Error Message     | Output | String    | Y        | Stores error messages.  |
| ReturnVal         | Output | String    | Y        | Returns Success or Fail. Not used.  |

## Pseudo-Algorithm for DiscardCart

- 1 Get the Quote business component.
- 2 Get the Id of the current record in the Quote business component:  
If Id <> CurrentQuoteId profile attribute, go to step 3;  
Else go to step 4.
- 3 DiscardCart is being invoked from the eSales - Edit Quote Process, eSales - Order Quote Process, eSales - Modify Order Process, or eSales - Reorder Process workflow. The current record in the Quote business component is the quote being acted upon, and it is not the current quote in the Shopping Cart.  
  
Get another instance of the Catalog business object and the Quote business component, and position the second instance of the Quote business component on the quote record as indicated by the CurrentQuoteId profile attribute. As the first instance of the Quote business component is not being acted upon, from now on Quote means the second instance of the Quote business component.
- 4 Set the CurrentQuoteId profile attribute = EMPTY\_CART.
- 5 Check the field value of [Quote Type]:  
If it is Persistent Cart or Anonymous Quote, then delete the record.

## OrderQuote Method

OrderQuote is called in the eSales - Checkout Process workflow. It checks if there are any line items in the Shopping Cart. It returns an error if the Shopping Cart does not have any line items. It also sets the CurrentCheckOutQuoteId profile attribute.

### Preconditions for Calling the OrderQuote Method

This method is invoked in the eSales - Checkout Process workflow. The sequence called is OrderQuote, then ValidQuote (see [“ValidQuote Method” on page 136](#)), and then RemoveInvalidItems (see [“RemoveInvalidItems Method” on page 137](#)).

### Arguments for the OrderQuote Method

The OrderQuote method has the arguments shown in [Table 32](#).

Table 32. OrderQuote Arguments

| Name              | Type   | Data Type | Optional | Comments  |
|-------------------|--------|-----------|----------|---|
| Return Error Code | Input  | String    | Y        | If equal to N, the method will not return errors.<br><br>When a method returns an error, the workflow stops. To avoid stopping the workflow, you can set this input to N. This way the Shopping Service will not return an error, but instead save the error message in the output argument Error Message. You can then handle the error message as you want. |
| Error Message     | Output | String    | Y        | Stores error messages.  |
| Quote Id          | Input  | String    | Y        | Not used.   |
| ReturnVal         | Output | String    | Y        | Returns Success or Fail. Not used.  |

### Pseudo-Algorithm for OrderQuote

- 1 Get the Quote and Quote Item business components.
- 2 If there are any records in the Quote Item business component, go to step 3;  
Else go to step 4.
- 3 Set the CurrentCheckOutQuoteId profile attribute.
- 4 Set the Error Message output argument if there is any error;  
If Return Error Code = N, do not return an error;  
Else return an error.

## ValidQuote Method

ValidQuote is called in the eSales - Checkout Process workflow to validate the line items in the Shopping Cart. Valid line items are those that do not have an empty Base Price field in the Quote Item business component.

ValidQuote sets the Invalid output argument to Y if there are any line items that are invalid, and also sets Invalid Flag field for each line item that is invalid. The Invalid Flag field is a virtual field in the Quote Item business component that is persistent in the session.

### Preconditions for Calling the ValidQuote Method

- This method is invoked in the eSales - Checkout Process workflow. The sequence called is OrderQuote (see [“OrderQuote Method” on page 135](#)), then ValidQuote, and then RemoveInvalidItems (see [“RemoveInvalidItems Method” on page 137](#)).
- This method checks only the Base Price field in the Quote Item business component. If this field is empty for a record, then the record is invalid. You can override this method to do your own validity checking.

### Arguments for the ValidQuote Method

The ValidQuote method has the arguments shown in [Table 33](#).

Table 33. ValidQuote Arguments

| Name              | Type   | Data Type | Optional | Comments  |
|-------------------|--------|-----------|----------|---|
| Return Error Code | Input  | String    | Y        | If equal to N, the method will not return errors.<br><br>When a method returns an error, the workflow stops. To avoid stopping the workflow, you can set this input to N. This way the Shopping Service will not return an error, but instead save the error message in the output argument Error Message. You can then handle the error message as you want. |
| Error Message     | Output | String    | Y        | Stores error messages.  |
| Invalid           | Output | String    | Y        | Returns Y if there are any invalid items in the Shopping Cart. Else returns N.  |
| Quote Id          | Input  | String    | Y        | Not used.   |
| ReturnVal         | Output | String    | Y        | Returns Success or Fail. Not used.  |

### Pseudo-Algorithm for ValidQuote

- 1 Get the Quote and Quote Item business component. Create a variable invalidFlag=N.



- 2 For each record in the Quote Item business component, check the Base Price field:  
If this field is empty, set invalidFlag = Y and set the Invalid Flag field = Y for that record.
- 3 Set the Invalid output argument = invalidFlag.
- 4 Set the Error Message output argument if there is any error;  
If Return Error Code = N, do not return an error;  
Else return an error.

## RemoveInvalidItems Method

RemoveInvalidItems is called in Check Out workflow. It deletes each record in the Quote Item business component whose Invalid Flag is Y.

### Preconditions for Calling the RemoveInvalidItems Method

- This method is invoked in the eSales - Checkout Process workflow. The sequence called is OrderQuote (see [“OrderQuote Method” on page 135](#)), then ValidQuote (see [“ValidQuote Method” on page 136](#)), and then RemoveInvalidItems.
- This method checks only the Base Price field in the Quote Item business component. If this field is empty for a record, then the record is invalid. You can override this method to do your own validity checking.

### Arguments for the RemoveInvalidItems Method

The RemoveInvalidItems method has the arguments shown in [Table 34](#).

Table 34. RemoveInvalidItems Arguments

| Name              | Type   | Data Type | Optional | Comments  |
|-------------------|--------|-----------|----------|---|
| Return Error Code | Input  | String    | Y        | If equal to N, the method will not return errors.<br><br>When a method returns an error, the workflow stops. To avoid stopping the workflow, you can set this input to N. This way the Shopping Service will not return an error, but instead save the error message in the output argument Error Message. You can then handle the error message as you want. |
| Error Message     | Output | String    | Y        | Stores error messages.  |
| Invalid           | Input  | String    | Y        | If equal to N, do nothing.<br><br>Else loop through each line item to delete invalid items.   |

Table 34. RemoveInvalidItems Arguments

| Name      | Type   | Data Type | Optional | Comments                           |
|-----------|--------|-----------|----------|------------------------------------|
| Quote Id  | Input  | String    | Y        | Not used.                          |
| ReturnVal | Output | String    | Y        | Returns Success or Fail. Not used. |

## Pseudo-Algorithm for RemoveInvalidItems

- 1 Get the Quote and Quote Item business components.
- 2 Check the Invalid input argument:  
If Invalid = N, go to step 4.
- 3 For each record in the Quote Item business component, check the Invalid Flag field:  
If the field value is Y, delete the record.
- 4 Set the Error Message output argument if there is any error;  
If Return Error Code = N, do not return an error;  
Else return an error.

## CreateOrder Method

CreateOrder method creates an order from the current quote. It creates a record in the Order Entry - Orders business component. Also, it copies records from the child business component Quote Item to Order Entry - Line Items and from the grandchild business component Quote Item XA to Order Item XA.

### Preconditions for Calling the CreateOrder Method

Field maps are used to copy fields from the Quote to Order Entry - Orders, Quote Item to Order Entry - Line Items, and Quote Item XA to Order Item XA business components.

Sample field maps are shown in [Table 35](#). See the Shopping Service business service user properties in Siebel Tools for more details.

Table 35. Sample Field Maps for the CreateOrder Method

| Name                                     | Value                                       |
|--|---|
| Quote Order FieldMap 100                 | Contact Id Contact Id                       |
| Quote Item Order Item FieldMap 100       | Class Registration Id Class Registration Id |
| Quote Item XA Order Item XA FieldMap 101 | Name Name                                   |

## Arguments for the CreateOrder Method

The CreateOrder method has the arguments shown in [Table 36](#).

Table 36. CreateOrder Arguments

| Name              | Type   | Data Type | Optional | Comments  |
|-------------------|--------|-----------|----------|---|
| Return Error Code | Input  | String    | Y        | If equal to N, the method will not return errors.<br><br>When a method returns an error, the workflow stops. To avoid stopping the workflow, you can set this input to N. This way the Shopping Service will not return an error, but instead save the error message in the output argument Error Message. You can then handle the error message as you want. |
| Error Message     | Output | String    | Y        | Stores error messages.  |
| Quote Id          | Input  | String    | Y        | Not used.   |
| Order Id          | Output | String    | Y        | ID of the order record that is created.   |
| ReturnVal         | Output | String    | Y        | Returns Success or Fail. Not used.  |

## Pseudo-Algorithm for CreateOrder

- 1 Get the Quote, Quote Item, Quote Item XA, Order Entry - Orders, Order Entry - Line Items, and Order Item XA business components.

The Quote business component is already positioned on the correct record, as the CreateOrder method is row-sensitive.

- 2 Create a record in the Order Entry - Orders business component. Use Quote|Order FieldMap *n* to copy fields from the Quote business component to the Order Entry - Orders business component.

**NOTE:** Only fields that have been fetched from the database will be copied over. Fields fetched are fields that either have the property Force Active set to True or that are displayed in the applet or that have been activated in scripting with method BusComp.ActivateField(FieldName).

- 3 For each record in the Quote Item business component, copy it to the Order Entry - Line Items business component using the Quote Item|Order Item FieldMap *n* field maps. To maintain hierarchical relationships, copy the parent record before copying its child records. Also copy records from the grandchild business component Quote Item XA to Order Item XA.

**NOTE:** As Quote Item and Order Entry - Line Items have hierarchies, you cannot simply loop through the records in Quote Item and copy them to Order Entry - Line Items. You need to maintain the hierarchical relationship. To do this, you need to copy the parent record before you copy its child records.

In the Shopping Service implementation, records are first loaded from Quote Item and a tree in the memory is generated. The tree is then traversed from the root, so the parent record is copied before the child records.

- 4 When doing step 3, for each record in the Quote Item business component:
- 5 Set the CurrentOrderId profile attribute to the order that is created;  
Set the Order Id output argument;  
Clear the CurrentCheckoutQuoteId profile attribute;  
If the Id of the quote record equals the CurrentQuoteId profile attribute, clear the CurrentQuoteId profile attribute by setting it = EMPTY\_CART.
- 6 Set the Error Message output argument if there is any error;  
If Return Error Code = N, do not return an error;  
Else return an error.

## Reorder Method

Reorder is the reverse of CreateOrder. It creates a quote from an existing order.

### Preconditions for Calling the Reorder Method

- Field maps are used to copy fields from the Order to Quote and Order Entry - Line Items to Quote Item business components, in the reverse of the CreateOrder method. Field maps are used to copy records from the Order Item XA to the Quote Item XA business component.

Sample field maps are shown in [Table 37](#). See the Shopping Service business service user properties in Siebel Tools for more details.

Table 37. Sample Field Maps for the Reorder Method

| Name                     | Value                 |
|--------------------------|-----------------------|
| Quote Order FieldMap 100 | Contact Id Contact Id |

Table 37. Sample Field Maps for the Reorder Method

| Name  | Value   |
|---|---|
| Quote Item Order Item FieldMap 100                | Class Registration Id <br>Class Registration Id |
| Quote Template Item XA Quote Item XA FieldMap 100 | Attribute Id Attribute Id                       |

## Arguments for the Reorder Method

The Reorder method has the arguments shown in [Table 38](#).

Table 38. Reorder Arguments

| Name              | Type   | Data Type | Optional | Comments   |
|-------------------|--------|-----------|----------|--|
| Return Error Code | Input  | String    | Y        | If equal to N, the method will not return errors. When a method returns an error, the workflow stops. To avoid stopping the workflow, you can set this input to N. This way the Shopping Service will not return an error, but instead save the error message in the output argument Error Message. You can then handle the error message as you want. |
| Error Message     | Output | String    | Y        | Stores error messages.   |
| Quote Id          | Output | String    | Y        | ID of the quote that is created.   |
| Order Id          | Output | String    | Y        | Not used.  |
| ReturnVal         | Output | String    | Y        | Returns Success or Fail. Not used.   |

## Pseudo-Algorithm for Reorder

- 1 Get the Order Entry - Orders, Order Entry - Line Items, and Order Item XA business components. Get an instance of the Catalog business object and of the Quote, Quote Item, and Quote Item XA business components from this Catalog business object instance.  
  
The Order Entry - Orders business component is already positioned on the correct record, as the Reorder method is row-sensitive.
- 2 Create a record in the Quote business component. Use Quote|Order FieldMap n to copy fields from the Order Entry - Orders business component to Quote.

- 3 For each record in the Order Entry - Line Items business component, copy it to the Quote Item business component using Quote Item|Order Item FieldMap *n* field maps. To maintain hierarchical relationships, copy the parent record before copying its child records. Also, copy records from the grandchild business component Order Item XA to Quote Item XA.

**NOTE:** As Order Entry - Line Items and Quote Item have hierarchies, you cannot simply loop through the records in Order Entry - Line Items and copy them to Quote Item. You need to maintain the hierarchical relationship. To do this, you need to copy the parent record before you copy its child records.

In the Shopping Service implementation, records are first loaded from Order Entry - Line Items and a tree in the memory is generated. The tree is then traversed from the root, so the parent record is copied before the child records.

- 4 When doing step 3, for each record in the Order Entry - Line Items business component:
- 5 Set the Error Message output argument if there is any error;
  - If Return Error Code = N, do not return an error;
  - Else return an error.

## GetShoppingCartStatus Method

GetShoppingCartStatus is called to get the status of the Shopping Cart (current quote as indicated by the CurrentQuoteId profile attribute). It returns three output arguments—Status, Quote Type, and Count.

### Preconditions for Calling the GetShoppingCartStatus Method

None.

### Arguments for the GetShoppingCartStatus Method

The GetShoppingCartStatus method has the arguments shown in [Table 39](#).

Table 39. GetShoppingCartStatus Arguments

| Name       | Type   | Data Type | Optional | Comments   |
|------------|--------|-----------|----------|--|
| Quote Type | Output | String    | Y        | Type of the current quote record.<br><br>If Persistent Cart, check Status and Count to determine whether the quote is empty or not.<br><br>If not Persistent Cart, Status will always be Empty and Count will always be 0 even though the current quote record might contain line items. |
| Status     | Output | String    | Y        | Can be Empty or Not Empty.   |

Table 39. GetShoppingCartStatus Arguments

| Name      | Type   | Data Type | Optional | Comments                                   |
|-----------|--------|-----------|----------|--|
| Count     | Output | String    | Y        | Number of line items in the Shopping Cart. |
| Quote Id  | Input  | String    | Y        | Not used.                                  |
| ReturnVal | Output | String    | Y        | Returns Success or Fail. Not used.         |

## Pseudo-Algorithm for GetShoppingCartStatus

- 1 If CurrentQuoteId is empty or equals to EMPTY\_CART:
  - Set Quote Type = Quotation;
  - Set Status = Empty;
  - Set Count = 0;
  - End.
- 2 If Quote Type is not Persistent Cart:
  - Set Quote Type accordingly;
  - Set Status = Empty;
  - Set Count = 0;
  - End.
- 3 Loop through the Quote Item business component. Set Status and Count accordingly.

## MoveToCart Method

MoveToCart moves a quote record into the Shopping Cart. By doing this, it sets the CurrentQuoteId profile attribute to the quote record ID. It also sets the Price List Id field value of this quote to the current price list for the session as indicated by the PriceListId profile attribute, for users associated with an account that has a default price list, and by DefaultPriceListId for users who are not associated with such an account. This triggers a reprice if the price list for this quote has been changed.

In addition, if the field value of Quote Type is not Auction Quote, it changes the quote type to Persistent Cart if the application supports persistent Shopping Carts. For applications that do not support Shopping Carts, the Shopping Service does not change the quote type.

## Preconditions for Calling the MoveToCart Method

- This method is invoked in the following workflows:
  - eSales - Edit Quote Process
  - eSales - Order Quote Process

- eSales - Modify Order Process
- eSales - Reorder Process
- The field value of Quote Type for the current quote record can be Persistent Cart, Auction Quote, or Quotation. When the quote type is not Auction Quote, if the application supports persistent Shopping Carts, the quote type is changed to Persistent Cart.

## Arguments for the MoveToCart Method

The MoveToCart method has the arguments shown in [Table 40](#).

Table 40. MoveToCart Arguments

| Name      | Type   | Data Type | Optional | Comments                           |
|-----------|--------|-----------|----------|------------------------------------|
| Quote Id  | Input  | String    | Y        | Not used.                          |
| ReturnVal | Output | String    | Y        | Returns Success or Fail. Not used. |

## Pseudo-Algorithm for MoveToCart

- 1 Set the CurrentQuoteId profile attribute to the quote record ID. As the MoveToCart method is row-sensitive, the current record in the Quote business component is the record to move to the Shopping Cart.
- 2 If the PriceListId profile attribute is not empty and if the Price List Id field in this quote record <> PriceListId profile attribute, then change the Price List Id for this quote. This triggers a reprice. Note that you need to use DefaultPriceListId instead of PriceListId if the user is not associated with an account that has a default price list.
- 3 If Quote Type is not Persistent Cart or Auction Quote, change it to Persistent Cart.

## RepriceQuote Method

RepriceQuote reprices the quote.

## Preconditions for Calling the RepriceQuote Method

None.



## Arguments for the RepriceQuote Method

The RepriceQuote method has the arguments shown in [Table 41](#).

Table 41. RepriceQuote Arguments

| Name              | Type   | Data Type | Optional | Comments                                |
|-------------------|--------|-----------|----------|---|
| Quote Id          | Input  | String    | Y        | ID for the quote record to be repriced. |
| Error Message     | Output | String    | Y        | Error message, if any.                  |
| Return Error Code | Input  | String    | Y        | Does not return an error if equal to N. |
| ReturnVal         | Output | String    | Y        | Returns Success or Fail. Not used.      |

## Pseudo-Algorithm for RepriceQuote

- 1 Position the Quote business component on the record indicated by the Quote Id input argument.
- 2 Call Pricing Manager to get the current price list ID:  
If this price list ID is different from the quote record's price list Id, then change the quote record's price list ID.

## SelectPrimary Method

This method is used to select a row from the picklist.

## Preconditions for Calling the SelectPrimary Method

None.

## Arguments for the SelectPrimary Method

The SelectPrimary method has the arguments shown in [Table 42](#).

Table 42. SelectPrimary Arguments

| Name             | Type   | Data Type | Optional | Comments                                 |
|------------------|--------|-----------|----------|--|
| PickList Field   | Input  | String    | Y        | Name of the field that has the picklist. |
| Primary ID Field | Input  | String    | Y        | Primary Id field.                        |
| Primary Row Id   | Input  | String    | Y        | Row ID for the picklist.                 |
| ReturnVal        | Output | String    | Y        | Returns Success or Fail. Not used.       |

## Pseudo-Algorithm for SelectPrimary

- 1 Get and check the input arguments.
- 2 Get the picklist business component from the Quote business component with the PickList Field input argument.
- 3 Set the search specification using either the Primary Row Id or Primary ID Field input argument.
- 4 Pick the first row if an active row exists;  
Else clear the picklist.
- 5 Write and refresh the business component.

## SetBusCompFields Method

This method is used to set a list of fields in a business component.

### Preconditions for Calling the SetBusCompFields Method

None.

### Arguments for the SetBusCompFields Method

The SetBusCompFields method has the arguments shown in [Table 43](#).

Table 43. SetBusCompFields Arguments

| Name               | Type  | Data Type | Optional | Comments                                     |
|--------------------|-------|-----------|----------|--|
| Business Object    | Input | String    | N        | Business object.                             |
| Business Component | Input | String    | N        | Business component.                          |
| Search Field       | Input | String    | Y        | Field on the business component.             |
| Search Value       | Input | String    | Y        | Value to be set to the search specification. |
| <Field Name1>      | Input | String    | Y        | Name/value pair.                             |
| <Field Name2>      | Input | String    | Y        | Name/value pair.                             |

### Pseudo-Algorithm for SetBusCompFields

- 1 Get the required inputs to get an instance of the business component.
- 2 Get the search specification fields:  
If they are passed, then set the search specification on the business component.
- 3 Query the business component if it has not been queried or if the search specification is passed.

- 4 Go through all the input arguments for <Field Name *n*> and set the value to the business component.
- 5 Write to the record.

## ValidatePayment Method

This method is used to validate the payment information for either a purchase order or a credit card.

### Preconditions for Calling the ValidatePayment Method

None.

### Arguments for the ValidatePayment Method

The ValidatePayment method has the arguments shown in [Table 44](#).

Table 44. ValidatePayment Arguments

| Name               | Type   | Data Type | Optional | Comments                                |
|--------------------|--------|-----------|----------|---|
| PO Number          | Input  | String    | N        | Purchase order number.                  |
| Credit Card Number | Input  | String    | Y        | Credit card number.                     |
| Credit Card Type   | Input  | String    | Y        | Credit card type.                       |
| Expiration Month   | Input  | String    | Y        | Expiration month.                       |
| Expiration Year    | Input  | String    | Y        | Expiration year.                        |
| Return Error Code  | Input  | String    | N        | Does not return an error if equal to N. |
| Error Message      | Output | String    | N        | Error message, if any.                  |

**NOTE:** If you want to perform credit card validation on a business component other than Quote, the business component must be either based on the CSSBCBase class, based on a class that is itself based on the CSSBCBase class, or based on a class that inherits from the CSSBCBase class, such as CSSBCQuote. For more information, see the section on credit card user properties in *Using Siebel Tools*.

### Pseudo-Algorithm for ValidatePayment

- 1 Check what kind of payment is entered by checking the PO Number and Credit Card Number input arguments.
- 2 If PO Number has a value, then the payment information is valid.
- 3 If Credit Card Number has a value, then get all the required fields.

- 4 Pass the credit card information to the Credit Card Transaction Service business service to validate the information.

## ValidateAddress Method

This method is used to validate the address information by checking to see that the necessary information has been entered. Input argument names are used in any error messages.

### Preconditions for Calling the ValidateAddress Method

None.

### Arguments for the ValidateAddress Method

The ValidateAddress method has the arguments shown in [Table 45](#).

Table 45. ValidateAddress Arguments

| Name              | Type   | Data Type | Optional | Comments                              |
|-------------------|--------|-----------|----------|---------------------------------------|
| Street Address    | Input  | String    | N        |                                       |
| State             | Input  | String    | N        |                                       |
| City              | Input  | String    | N        |                                       |
| Country           | Input  | String    | N        |                                       |
| Postal Code       | Input  | String    | N        |                                       |
| Return Error Code | Input  | String    | N        | Do not return an error if equal to N. |
| Error Message     | Output | String    | N        | Error message, if any.                |

### Pseudo-Algorithm for ValidateAddress

Go through all the input arguments and make sure that the values are not null.

## SetPendingRequest Method

This method is used to set the Profile attribute *Pending Request*, which is used to continue an operation after login.

#### Arguments:

- SWEApplet
- SWECmd

- SWEMethod
- SWENeedContext
- SWEReqRowId
- SWEService
- SWEShowPopup
- SWEView

**Usage:**

These arguments are used to setup pending requests to Siebel Web Engine.



# Index

## A

**access groups, defined and relationship to user** 17

### accounts

associating price lists with 52  
associating users with 52  
auto-approval limit, setting 53  
creating 52  
credit auto-approval limit, setting 53  
credit checks for, skipping 53  
use of purchase orders 53

### AddToCart method

about 112  
arguments 118  
preconditions for calling 113  
pseudo-algorithm 119  
pseudo-algorithm for DefaultAddToCart 119  
pseudo-algorithm for PostAddToCart 119

### AddToFavoritesList method

about 120  
arguments 121  
preconditions for calling 120  
pseudo-algorithm 121

### administrative tasks

account credit auto-approval limit, setting 53  
anonymous Shopping Carts, managing 51  
credit checks for an account, skipping 53  
customer account, associating price lists with 52  
customer account, associating users with 52  
customer account, creating 52  
customer account, setting credit auto-approval limit 53  
customer account, setting purchase order auto-approval limit 53  
quote expiration dates, setting 51

**anonymous quote type** 57

**anonymous Shopping Carts, managing** 51

**applets, hiding** 20

### application

about 11  
configuring 19  
customizing 14  
features 11  
process of setting up 13

**Approve Order Process workflow** 98

**auctions, Silent Authorize Order workflow** 108, 109

**Authorize and Settle credit card charge modification, about** 55

**Authorize credit card charge modification, about** 54

**Authorize Order Process workflow** 87, 89  
**auto-approval limit, setting for an account** 53

## C

**CalculateShippingCost script, about** 23

**Cancel Order Process workflow** 100

**Catalog page, home page link** 12

**Check Shipment button, about using checkout** 45

Authorize Order Process workflow 87, 89  
Checkout Process workflow 76  
Complete Checkout Process workflow 86  
Credit Check-Quotes workflow 77  
Current User Type workflow 95, 96  
Enter Payment Details Process workflow 82, 84  
Enter Shipping Details Process workflow 78  
Save Account Shipping Address workflow 92  
Save Credit Card workflow 90  
Save Shipping Address workflow 91, 92  
Set Primary Account Address workflow 93  
Setting PO Status workflow 94  
workflows list 74

**communications drivers, about setting up** 25

**Complete Checkout Process workflow** 86

**conditional expressions, used to hide applets** 20

### contacts

about using purchase orders 53  
allowing to purchase orders 54

### CreateOrder method

about 138  
arguments 139  
preconditions for calling 138  
pseudo-algorithm 139

**credit auto-approval limit for an account, setting** 53

### credit card charges

credit card authorization, about querying

- for 55
    - credit card authorization, querying for Siebel
      - Response messages 56
    - modifications allowed 54
    - modifying, fields 54
    - reversing charges 55
  - Credit Card Transaction Service**
    - business service methods, list 29
    - business service user properties 31
  - Credit Card Transaction Service, business service methods**
    - AuthCharge 29
    - Authorization 29
    - Charge 30
    - Process 30
    - Refund 30
    - Reverse 30
    - Validate 30
  - credit card validation, on business component other than quote** 147
  - credit cards**
    - data, about encrypting and eSales fields 24
    - payments, workflow for authorizing 87
    - transaction service, calling from other business components 42
    - transaction service, calling from other places in eSales 42
    - transactions, data maps used in 37
    - transactions, enabling with CyberSource 28
  - Credit Check-Quotes workflow** 77
  - Current User Type workflow** 95, 96
  - customer accounts**
    - auto-approval limit, setting 53
    - creating an account 52
    - credit auto-approval limit, setting 53
    - price lists, associating with 52
    - users, associating with 52
  - CyberSource** 27
    - accessing from eSales 27
    - credit card transactions, enabling 28
    - customizing integration 40
    - downloading latest version 27
    - installing 27
    - integration with Siebel eSales 28
    - modifying 42
    - Web site 27
  - CyberSource Adapter Service**
    - about and methods 38
    - scripts, examining 39
    - scripts, list 39
    - scripts, using to modify 40
  - CyberSource, customizing**
    - Credit Card Transaction Service, calling from other business components 42
    - Credit Card Transaction Service, calling from other places in eSales 42
    - fields not in Siebel database, passing data from 41
    - fields not in Siebel database, passing data to 41
    - scripts, modifying in CyberSource Adapter Service 40
    - third-party payment applications, using 42
  - CyberSource, integrating with eSales**
    - address verification, about 40
    - business service methods 29
    - business service user properties, table of 31
    - CyberSource Adapter Service, methods, user properties, and scripts 38
- D**
- data maps**
    - CyberSource, about using and viewing 37
    - CyberSource, used in credit card transactions 37
  - DAVSNO reply code, returned by CyberSource** 40
  - (declarations) script, about** 23
  - DefaultAddToCart, using pseudo-algorithm** 119
  - Delete Old Anonymous Quotes workflow** 74
  - DiscardCart method**
    - about 133
    - arguments 134
    - preconditions for calling 133
    - pseudo-algorithm 134
  - Display Error Process workflow** 103
- E**
- Edit Quote Process workflow** 68
  - email notification, customizing**
    - about and tasks involved 25
    - communications driver profiles, about setting up 25
    - email template, modifying 25
    - workflows, using to control notification 25
  - email notifications workflows**
    - about 66
    - list 96
  - Enter Payment Details Process workflow** 82, 84
  - Enter Shipping Details Process workflow** 78
  - error messages**
    - credit card authorizations, about querying for 55
    - credit card authorizations, querying for Siebel Response message 56



Display Error Process workflow 103  
workflows for displaying 103

## eSales

See Siebel eSales

## eSales home page

See home page

## External Shipment Tracking ASI user properties, setting 47

## F

### favorites list

making items unavailable 12  
preventing item orders 12

## G

### GetShippingMethodCost script, about 23

### GetShippingRate script, about 23

### GetShoppingCartStatus method

about 142  
arguments 142  
preconditions for calling 142  
pseudo-algorithm 143

## H

### home page

about and screen example 12  
hyperlinks and tabs 12

## I

### installing

Cybersource 27  
required Siebel Business Applications 14

### integration

customizing for CyberSource 40

### integration objects

about 33  
authorization integration components (output) 34  
authorize and settle (CyberSource Input) 36  
authorize and settle (CyberSource Output) 36  
configuration user properties 33  
CreditCard authorization (CyberSource Input) 34  
CreditCard authorization (CyberSource Output) 34  
CreditCard Payments authorization 34  
CreditCard-Quote authorization (Siebel Input) 35  
CreditCard-Quote authorization (Siebel Output) 35  
refund (CyberSource Input) 36

refund (CyberSource Output) 37  
settle-charge (CyberSource Input) 35  
settle-charge (CyberSource Output) 36  
using to set up shipping charge calculation 22

## L

### Login View workflow 105

### LOV (List of Values) cache, clearing 51

## M

### methods

AddToCart 112  
AddToFavoriteList 120  
CreateOrder 138  
DiscardCart 133  
GetShoppingCartStatus 142  
MoveToCart 143  
OrderQuote 135  
OrderTemplate 126  
RemoveInvalidItems 137  
Reorder 140  
RepriceQuote 144  
SaveCartAsQuote 131  
SaveCartAsTemplate 129  
SelectPrimary 145  
SetApplicationName 124  
SetBusCompFields 146  
SetShoppingCart 122  
ValidateAddress 148  
ValidatePayment 147  
ValidQuote 136

### methods, about

invoked by workflows 128  
user action methods list 111

### Modify Order Process workflow 101

### MoveToCart method

about 143  
arguments 144  
preconditions for calling 143  
pseudo-algorithm 144

### My Account, home page link 12

### My Company Applet, hiding 20

## N

### notification, customizing

about and tasks involved 25  
communications driver profiles, about setting up 25  
email templates, modifying 25  
email workflows, list 96  
workflows, about using to control notification 25

**O****object manager parameters**

- default parameters, viewing 62
- Mobile Web Client, modifying parameters 64
- Web client, modifying parameters 63

**Object Manager, support for shopping cart** 62**Order Confirmation View workflow** 106, 107**order modification, workflows** 100**Order Quote Process workflow** 70**order workflows**

- Approve Order Process workflow 98
- auction quotes to orders 108
- Authorize Order Process 87, 89
- Cancel Order Process workflow 100
- Checkout Process 76
- Complete Checkout Process 86
- Credit Check-Quotes 77
- Current User Type 95, 96
- email workflows list 96
- Enter Payment Details Process 82, 84
- Enter Shipping Details Process 78
- error messages 103
- features 66
- Modify Order Process 101
- modifying processes 65
- Process Designer, accessing 65
- Reorder Process workflow 102
- Save Account Shipping Address 92
- Save Cart Process workflow 72
- Save Credit Card 90
- Save Shipping Address 91, 92
- Set Primary Account Address 93
- Setting PO Status 94

**OrderQuote method**

- about 135
- arguments 135
- preconditions for calling 135
- pseudo-algorithm 135

**orders, modifying**

- See email notification, customizing

**OrderTemplate method**

- about 126
- arguments 128
- preconditions for calling 127
- pseudo-algorithm 128

**P****package sorting**

- order, changing 49
- turning off 49

**Package Tracking Number, defined** 45**package, defined** 45**payment applications, using third-party persistent cart quote type, about and process overview** 57**personalization**

- about 19
- applets, hiding 20

**portal pages, including unstructured content** 15**PostAddToCart method, using pseudo-algorithm** 119**price lists**

- customer accounts, associating with 52
- default, setting 18
- row number, finding 18
- Set PriceList WebLogin action, about 61
- setting up 18

**Process Designer, accessing** 65**purchase approval, workflows** 97**purchase orders**

- Approve Order Process workflow 98
- Reject Order Process workflow 99
- use by accounts 53
- use by contacts 53, 54
- workflow for authorizing 87

**Q****quotation quote type** 58**Quote Empty Form Applet, hiding with conditional expression** 20**Quote Form Applet - Current, hiding** 20**Quote Item List Applet, hiding** 20**quotes**

- Delete Old Anonymous Quotes workflow 74
- Edit Quote Process workflow 68
- expiration dates, setting 51
- Order Quote Process workflow 70
- quote types 57
- Reprice Process workflow 71
- workflows 67

**R****Recommended Product List Applet, hiding** 20**Refund credit card charge modification, about** 55**Reject Order Process workflow** 99**RemoveInvalidItems method**

- about 137
- arguments 137
- preconditions for calling 137
- pseudo-algorithm 138

**Reorder method**

- about 140

- arguments 141
- preconditions for calling 140
- pseudo-algorithm 141
- Reorder Process workflow** 102
- Reprice Process workflow** 71
- RepriceQuote method**
  - about 144
  - arguments 145
  - preconditions for calling 144
  - pseudo-algorithm 145
- responsibilities**
  - and user types 17
  - predefined, list 16
  - setting up 16
  - Web anonymous user 16
  - Web corporate user 16
  - Web delegated customer administrator 17
  - Web purchasing manager 16
  - Web registered user 16
- returned data, extending** 48
- reversing charges** 55
- RSA encryption, about backing up data** 24
- run-time events, configuring Shopping Cart behavior** 60, 62

## S

- Save Account Shipping Address workflow** 92
- Save Cart Process workflow** 72
- Save Credit Card workflow** 90
- Save Shipping Address workflow** 91, 92
- SaveCartAsQuote method**
  - about 131
  - arguments 132
  - preconditions for calling 131
  - pseudo-algorithm 132
- SaveCartAsTemplate method**
  - about 129
  - arguments 130
  - preconditions for calling 129
  - pseudo-algorithm 130
- scripts**
  - business service scripts, setting up shipping charge calculation 23
  - CyberSource Adapter Service, using to modify scripts 40
- SelectPrimary method**
  - about 145
  - arguments 145
  - preconditions for calling 145
  - pseudo-algorithm 146
- Send Order Accepted Email (eSales) workflow** 96
- Send Order Awaiting Approval Email (eSales) workflow** 96
- Send Order Failed Email (eSales) workflow** 96
- Send Order Rejected Email (eSales) workflow** 96
- Send Order Shipped Email (eSales) workflow** 96
- Set Primary Account Address workflow** 93
- SetApplicationName method**
  - about 124
  - arguments 124
  - preconditions for calling 124
  - pseudo-algorithm 124
- SetApplicationName WebSessionStart action, about** 61
- SetBusCompFields method**
  - about 146
  - arguments 146
  - preconditions for calling 146
  - pseudo-algorithm 146
- SetPriceList WebLogin action, about** 61
- SetShoppingCart method**
  - about 122
  - arguments 122
  - preconditions for calling 122
  - pseudo-algorithm 123
- SetShoppingCart WebLogin action, about** 61
- Setting PO Status workflow** 94
- Settle credit card charge modification, about** 54
- setup procedures**
  - required for shipment tracking 46
- Shipment Identification Number**
  - about 45
  - defined 45
- Shipment Reference Number, defined** 45
- shipment tracking**
  - about third-party applications 45
  - required setup procedures 46
- Shipment Tracking Interface**
  - Request, about 45
  - Response, about 45
- Shipment Tracking Number, defined** 45
- shipment tracking setup procedures**
  - required 46
- Shipment Tracking-Package Tracking user properties, setting** 46
- Shipment, defined** 45
- Shipping Calculation Process workflow** 21
- shipping charge calculation, setting up**
  - about 20
  - business service scripts, list 23

- business services, arguments 21
  - integration objects, about using and list 22
  - modifying shipping charge calculation 24
  - workflow 21
  - Shipping Cost Service business service, scripts** 23
  - shipping tracking**
    - External Shipment Tracking ASI user
      - properties, setting 47
    - package sorting order, changing 49
    - package sorting, turning off 49
    - returned data, extending 48
    - Shipment Tracking-Package Tracking user
      - properties, setting 46
    - terms, list 45
    - Web services, configuring 47
  - ShippingCost - Quote (Siebel Input), integration object component** 23
  - ShippingCost - Quote (Siebel Output), integration object components** 23
  - Shopping Cart**
    - about 57
    - customizing 58
    - default object manager parameters,
      - viewing 62
    - home page link 12
    - object manager parameters, modifying in the
      - Web client 63
    - quote types 57
    - run-time events, about configuring 60
    - Save Cart Process workflow 72
    - Shopping Cart View workflow 108
    - Shopping Service business service method,
      - invoked by WebLogin 61
    - Shopping Service business service method,
      - invoked by WebSessionStart 61
    - Shopping Service business service
      - methods 58
    - support by Object Manager 62
  - Shopping Cart View workflow** 108
  - Shopping Service API**
    - about 111
    - User Action methods 111
    - workflow methods 128
  - Shopping Service business service**
    - customizing 58
    - methods 58
    - methods, detailed description 111
    - methods, User Action 111
    - methods, workflow 128
    - WebLogin, invoked by 61
    - WebSessionStart, invoked by 61
  - Siebel Business Applications, about installing** 14
  - Siebel eSales**
    - about 11
    - configuring 19
    - credit card encryption 24
    - customizing 14
    - features 11
    - integration with CyberSource 28
    - process of setting up 13
    - upgrading from earlier version 24
  - Siebel Response Messages** 55
  - Siebel Web Engine, about installing** 14
- T**
- templates**
    - email templates, modifying 25
    - preventing item orders 12
    - workflows 67
  - third-party payment applications**
    - See CyberSource
  - third-party shipping tracking applications**
    - See shipping tracking
  - third-party tax applications** 27
- U**
- unstructured content, creating** 15
  - user profile, SetApplicationName**
    - WebSessionStart action, about** 61
  - users**
    - access control tasks 15
    - action methods 111
    - associating with customer accounts 52
    - types and responsibilities 17
- V**
- Validate, Credit Card Transaction Service**
    - business method** 30
  - ValidateAddress method**
    - about 148
    - arguments 148
    - preconditions for calling 148
    - pseudo-algorithm 148
  - ValidatePayment method**
    - about 147
    - arguments 147
    - preconditions for calling 147
    - pseudo-algorithm 147
  - ValidQuote method**
    - about 136
    - arguments 136
    - preconditions for calling 136
    - pseudo-algorithm 136
  - views**
    - workflows for displaying 103

**W**

- Waybill number, defined** 45
- Web anonymous user, responsibility** 16
- Web corporate user, responsibility** 16
- Web delegated customer administrator**
  - new responsibility visible to, about adding 17
  - responsibility 17
- Web purchasing manager, responsibility** 16
- Web registered user, responsibility** 16
- Web services**
  - shipping tracking, configuring 47
- Web site**
  - CyberSource 27
  - multilingual, about deploying 15
- WebLogin run-time event, configuring**
  - Shopping Cart behavior** 62
- WebSessionStart run-time event, configuring**
  - Shopping Cart behavior** 60
- workflows**
  - activating 26
  - Anonymous Quote records, deleting 57
  - checkout 74
  - error message display 103
  - methods invoked 128
  - modifying processes 65
  - order modification 100
  - purchase approval 97
  - quote 67
  - Shipping Calculation Process 21
  - template 67
  - view display 103

