Siebel

eSales Administration

January 2020

Contents

ORACLE

ORACLE

Preface

This preface introduces information sources that can help you use the application and this guide.

Using Oracle Applications

To find guides for Oracle Applications, go to the Oracle Help Center at *http://docs.oracle.com/*.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the *Oracle Accessibility Program website*.

Contacting Oracle

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit *My Oracle Support* or visit *Accessible Oracle Support* if you are hearing impaired.

Comments and Suggestions

Please give us feedback about Oracle Applications Help and guides! You can send an email to: *oracle_fusion_applications_help_ww_grp@oracle.com*.

ORACLE

ORACLE

1 What's New in This Release

What's New in Siebel eSales Administration Guide, Siebel CRM 20.1 Update

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

What's New in Siebel eSales Administration Guide, Siebel CRM 19.1 Update

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

ORACLE

2 Siebel eSales Overview

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*
- *The Siebel eSales Web Site*

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.

ORACLE

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* .

ORACLE

3 **Setting Up Siebel eSales**

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*
- *Installing Siebel Business Applications*
- *Setting Up Order Management for Siebel eSales*
- *Customizing the eSales User Interface*
- *Creating Multilingual eSales Web Sites*
- *Controlling User Access to the eSales Web Site*
- *Configuring Siebel eSales*
- *Setting Up Personalization for Siebel eSales*
- *Setting Up Shipping Charge Calculations for Siebel eSales*
- *Encrypting Credit Card Data with RC2 Encryption*
- *Customizing Email Notifications for Siebel eSales*
- *Activating Workflows for Siebel eSales*

Process of Setting Up Siebel eSales

To set up Siebel eSales, perform the following tasks:

- *Installing Siebel Business Applications*.
- *Setting Up Order Management for Siebel eSales*
- *Customizing the eSales User Interface*.
- *Creating Multilingual eSales Web Sites*.
- *Controlling User Access to the eSales Web Site*.
- *Configuring Siebel eSales*.
- *Setting Up Personalization for Siebel eSales*.
- *Setting Up Shipping Charge Calculations for Siebel eSales*.
- *Encrypting Credit Card Data with RC2 Encryption*.
- *Customizing Email Notifications for Siebel eSales*.
- *Activating Workflows for Siebel eSales*.

ORACLE

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

- *Integrating Siebel eSales with Third-Party Payment Applications*
- *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 Siebel Application Interface, Gateway, 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 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*.

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

ORACLE

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*)
  - Catalogs and categories (see *Defining Access Groups*)
  - Products (see *Setting Up Price Lists for Siebel eSales*)
- 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

ORACLE

- 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 in the previous section. 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 the following table.

Job Role	Responsibilities	User Type
Web Anonymous User	Web Anonymous User	**Note:** No user User Type value is required.

ORACLE

Job Role	Responsibilities	User Type
Web Registered User	Web Registered User	Web Registered User
Web Corporate User	Web Corporate User	Web Corporate User
Web Purchasing Manager	• Web Corporate User • Web Purchasing Manager	Web Purchasing Manager
Web Delegated Customer Administrator	• Web Corporate User • Web Delegated Customer Administrator	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 Self Service 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 Web 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.

- Map the fields, controls, and applets to the placeholders in the appropriate template as follows:

ORACLE

- ○ 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).

- Save your changes and submit the workspace for delivery.

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 the following table are hidden depending on the application, the type of user, and whether or not the applets have any records in them.

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* .

ORACLE

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 the following image: Get data from Siebel, Calculate Shipping Cost, Write data into Siebel.
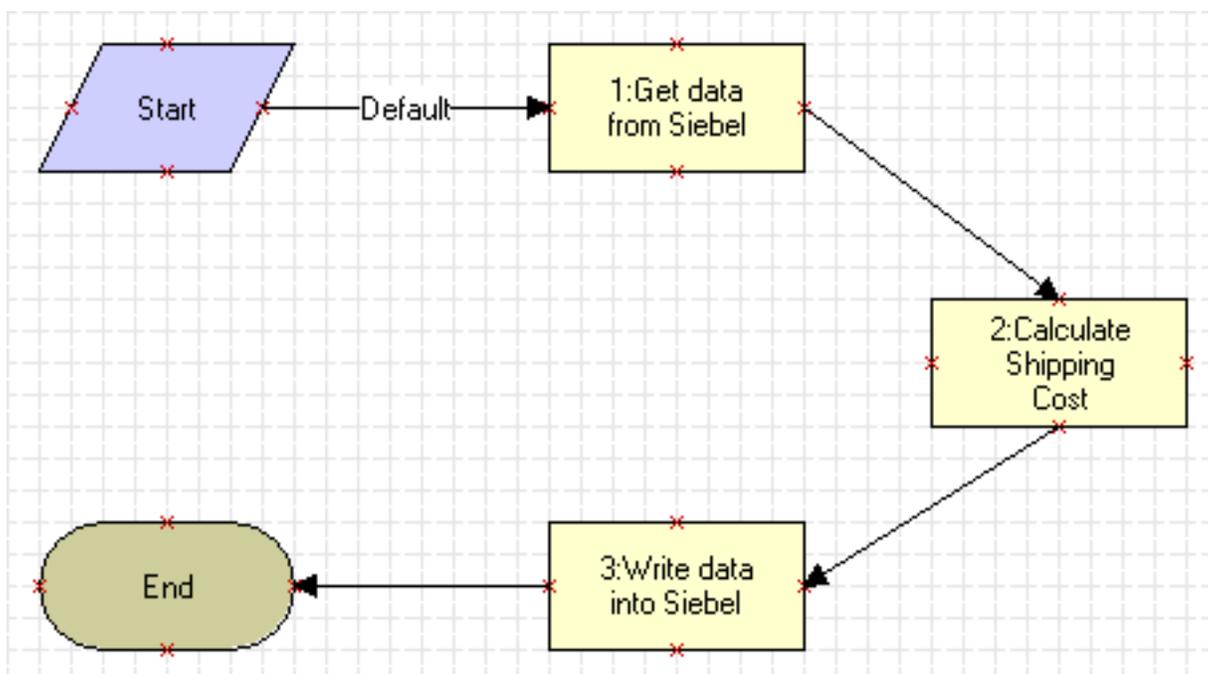
When the Shipping Calculation Process workflow is called by the eSales - Complete Checkout Process workflow (*eSales - Complete Checkout Process*), the following events happen:

- 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 the following table.

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 the following table.

Integration Component	Fields
Quote	Account Id
	Carrier Priority
	Currency Code
	Current Quote Ship To City

ORACLE

Integration Component	Fields
	Current Quote Ship To Country
	Current Quote Ship To Postal Code
	Current Quote Ship To State
	Quote Number
	Shipping Method
Quote Item	Line Total
	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* .

ORACLE

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 Siebel runtime repository 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 CRM are shown in the following table.

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

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.

ORACLE

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*.
- Modifying email workflows. See *Modifying Email Workflows*.
- Modifying email templates. See *Modifying Email Templates*.

For information on creating notifications, see *Siebel CTI 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 eSalesapplication. 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 CTI 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*. For more information about defining rules to enable automatic email notification, see *Siebel Business Process Framework: Workflow 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].

ORACLE

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 *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 Framework: Workflow Guide* .

ORACLE

4 Integrating Siebel eSales with Third-Party Payment Applications

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*
- *Installing CyberSource Software*
- *Enabling Credit Card Transactions with CyberSource*
- *CyberSource Integration with Siebel eSales*
- *Customizing Cybersource Integration*

About CyberSource Internet Commerce Suite

Siebel eSales retrieves credit card information from CyberSource as part of the eSales - Authorize Order Process workflow (*SS Authorize Order Process*) 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 Siebel Application Interface. 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.

ORACLE

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 following table.

Parameter	Value
CCAEnable	TRUE
CCAMerchantId	*merchant_id*
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)

ORACLE

- CyberSource Adapter Service
- Shopping Service

This section contains the following topics:

- *Credit Card Transaction Service Business Service*
- *Integration Objects*
- *Data Maps*
- *CyberSource Adapter Service Business Service*
- *Address Verification with Cyber Source*

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.

ORACLE

- ○ **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 the following table.

Business Service User Property	Value
Create Order on Web	1, 18, 19, 20, 21 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 *Administering Siebel eSales*

ORACLE

Business Service User Property	Value
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.
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.

Business Service User Property	Value
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.
	This response is given when CyberSource returns DAVSNO. For more information, see *Address Verification with Cyber Source*.
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 eSalesapplication.

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.

ORACLE

Configuration

CreditCard - Configuration (CyberSource) has the user properties shown in the following table.

User Property	Value	Comments
VendorDll	ics2api	Do not use the prefix and extension. 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." This is only used for the AuthCharge transaction.
merchant_id	ICS2Test	CyberSource varchar(20), case-sensitive. 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	None.
server_port	80	None.
timeout	90	In seconds. The default is 90.
decline_avs_flags	N	Comma-separated list of AVS codes that result in an AVS decline. For more information, see *Address Verification with Cyber Source*.

Authorization

- **CreditCard - Authorization (CyberSource Input).** Its integration component is Invoice, as shown in the following table.

Fields	Fields	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

ORACLE

Fields	Fields	Fields
• 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, as described in the following table.

Fields	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, as described in the following table.

Fields	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

ORACLE

Fields	Fields
• 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, as described in the following table.

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, as described in the following table.

Fields	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 Number	• Card Type
• Currency Code	• Email Address

ORACLE

Fields	Fields
• Home Phone	• Quote Number
• Work Phone	

Settle (Charge)

- **CreditCard - Charge (CyberSource Input).** Its integration component is Invoice, as described in the following table.

Fields	Fields
• Amount	• Auth Request Id
• Authorization Code	• Authorization Type
• Currency Code	• Duty Amount
• Merchant Descriptor	• Merchant Descriptor Contact
• Merchant Id	• National Tax
• Product Code	• Product Name
• 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 - Charge (CyberSource Output).** Its integration component is Invoice, as described in the following table.

Fields	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

ORACLE

Fields	Fields
• Charged Time	• Request Id
• Row Id	

Authorize and Settle (AuthCharge)

- **CreditCard - AuthCharge (CyberSource Output).** Its integration component is Invoice, as described in the following table.

Fields	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, as described in the following table.

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.

ORACLE

Refund

- **CreditCard - Refund (CyberSource Input).** Its integration component is Invoice, as described in the following table.

Fields	Fields	Fields
• Amount	• Bill Request Id	• Bill To City
• Bill To Country	• Bill To State	• Bill To Street Address
• Bill To Zipcode	• Card Expiration Month	• Card Expiration Year
• Card Holder Name	• Card Issue Number	• Card Number
• Card Start Month	• Card Start Year	• Currency Code
• Duty Amount	• Email Address	• Merchant Descriptor
• Merchant Descriptor Contact	• 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 - Refund (CyberSource Output).** Its integration component is Invoice, as described in the following table.

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

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

ORACLE

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 the following table.

Data Map	Source Integration Object	Target Integration Object
Authorization from Quote		
CreditCardAuthorization_ Quote2CyberSource	CreditCard - Quote - Authorization (Siebel Output)	CreditCard - Authorization (CyberSource Input)
CreditCardAuthorization_ CyberSource2Quote	CreditCard - Authorization (CyberSource Output)	CreditCard - Quote - Authorization (Siebel Input)
Authorization from Payments		
CreditCardAuthorization_ Payments2CyberSource	CreditCard - Payments	CreditCard - Authorization (CyberSource Input)
CreditCardAuthorization_ CyberSource2Payments	CreditCard - Authorization (CyberSource Output)	CreditCard - Payments
Charge from Payments		
CreditCardCharge_ Payments2CyberSource	CreditCard - Payments	CreditCard - Charge (CyberSource Input)
CreditCardCharge_ CyberSource2Payments	CreditCard - Charge (CyberSource Output)	CreditCard - Payments
AuthCharge from Quote		
CreditCardAuthorization_ Quote2CyberSource	CreditCard - Quote - Authorization (Siebel Output)	CreditCard - Authorization (CyberSource Input)
CreditCardAuthCharge_ CyberSource2Quote	CreditCard - AuthCharge (CyberSource Output)	CreditCard - Quote - AuthCharge (Siebel Input)
AuthCharge from Payments		

Data Map	Source Integration Object	Target Integration Object
CreditCardAuthorization_ Payments2CyberSource	CreditCard - Payments	CreditCard - Authorization (CyberSource Input)
CreditCardAuthCharge_ CyberSource2Payments	CreditCard - AuthCharge (CyberSource Output)	CreditCard - Payments
Refund from Payments		
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.

ORACLE

     ○   **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*
- *Passing Data from Other Fields in the Siebel Database*
- *Passing Data from Fields Not in the Siebel Database*
- *Calling the Credit Card Transaction Service from Other Places in Siebel eSales*
- *Calling the Credit Card Transaction Service from Other Business Components*
- *Using Other Third-Party Payment Applications*

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* .

ORACLE

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 Siebel runtime repository 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 Web 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 ICS2OBJS["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.

ORACLE

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* 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.

ORACLE

To create a new payment business service

1. In Web 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. Save the changed and submit the workspace for delivery.

ORACLE

5 Integrating Siebel eSales with Third-Party Shipment Tracking Applications

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*
- *Process of Setting Up Shipment Tracking*

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.

ORACLE

- **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*.
2. *Verifying the External Shipment Tracking ASI User Properties*.
3. *Configuring Web Services for Shipment Tracking*.

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*.

- *Changing the Package Sorting Order*.

- *Turning Off Package Sorting*.

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*.

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 described 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.

ORACLE

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*.

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 described 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*.

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 described in the following table.

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

ORACLE

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

Field	Comments
Name	Enter a name for the port.
Service Display Name	Enter External Shipment Tracking.
Transport	Select from the drop-down list. 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. 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. 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*.

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* .

ORACLE

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 the following table.

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*.

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*.

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.

ORACLE

6 Administering Siebel eSales

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*
- *Managing Anonymous Shopping Carts*
- *About Setting Up Customer Accounts*
- *Modifying Credit Card Charges*

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.

ORACLE

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 Framework: Workflow 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 Framework: Workflow 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* and the chapter on using accounts in Siebel Sales in *Siebel Applications Administration Guide* .

- Allowing contacts to use purchase orders.

ORACLE

For more information, see *About Allowing Accounts and Contacts to Use Purchase Orders* and *Allowing Contacts to Use Purchase Orders*.

- 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*.

Allowing Contacts to Use Purchase Orders

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

ORACLE

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*.

- **Identifying Errors**. See *Querying For Transactions With Error Messages*.

ORACLE

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 the following table.

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 *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.

ORACLE

7 Siebel eSales Shopping Cart

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*
- *Quote Types in Siebel eSales*
- *Shopping Service Business Service*
- *Run-Time Events in Siebel eSales*
- *Object Manager Support for Shopping Cart*

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.

ORACLE

- **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*
- *Shopping Service Business Service Methods*

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 the following table.

Method	Description	Used In Workflow
AddItem Method	Adds an item to the Shopping Cart or Favorites.	Not applicable.
CreateOrder Method	Converts a quote to an order. Also confirms training class and event registrations.	eSales – Authorize Order Process
DiscardCart Method	Deletes the current Shopping Cart and sets CurrentQuoteId to empty, but only when the Quote Type equals Persistent Cart.	eSales – Edit Quote Process eSales – Reorder Process
Get ShoppingCartStatus Method	Gets the status of the current Shopping Cart, which can be one of the following: - `Empty`. CurrentQuoteId is empty or Quote Type equals Persistent Cart. - `Not Empty`.	eSales – Edit Quote Process eSales – Modify Order Process eSales – Reorder Process

ORACLE

Method	Description	Used In Workflow
	• **No Save.** Also returns information such as Quote Item and Item Count.	
OrderQuote Method	Creates an order based on the quote.	eSales – Checkout Process eSales – Order Quote Process
Reorder Method	Creates a quote based on an existing order.	Not applicable.
RepriceQuote Method	Reprices quotes, for example when a customer orders a saved quote that has expired.	Not applicable.
SaveCartAsQuote Method	Saves the current Shopping Cart as a quote: • Finds the quote record whose Id equals CurrentQuoteId. • Saves it as a quote, that is, sets the quote status to Quotation. • Sets CurrentQuoteId to empty (empties the Shopping Cart).	eSales – Edit Quote Process eSales – Reorder Process eSales – Save Cart Process
SaveCartAsTemplate Method	Saves the current Shopping Cart as a template, but keeps the Shopping Cart. Ordering from a template copies the template items to the Shopping Cart.	eSales – Save Cart Process
SelectPrimary Method	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).	Not applicable.
SetApplicationName Method	Sets the ApplicationName dynamic user profile attribute. Triggered by the application's WebLogin event. Used in run-time events to support the different Shopping Cart behavior between Siebel eSales and other Siebel Business Applications.	Not applicable.
SetPendingRequest Method	Allows workflows to continue after anonymous users are sent to the Login view to register.	eSales – Checkout Process eSales – Save Cart Process
SetShoppingCart Method	Supports the persistent Shopping Cart and allows anonymous users to add items to the Shopping Cart. Triggered by the application's WebSessionStart event. When a session begins, the method does the following:	Not applicable.

ORACLE

Method	Description	Used In Workflow
	• If the user is anonymous, does nothing. • If CurrentQuoteId is not empty (that is, the user already has a Shopping Cart), checks if the Shopping Cart quote type is Anonymous Quote. If there is an Anonymous Quote, converts it to Persistent Cart and sets Contact Id to the login Id of the user. • 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. If there is no such quote, the user did not leave a Shopping Cart (for example, saved it as a quote or checked out).	

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*.

- **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 *SS Start Checkout Process*) 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*
- *Modifying Object Manager Parameters in the Web Client*

ORACLE

- *Modifying Object Manager Parameters in the Mobile Web Client*

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 move it to the next 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 the following table. In all cases, Condition is =.

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

ORACLE

Modifying Object Manager Parameters in the Mobile Web Client

For users of the Siebel Mobile Web Client, the application configuration files, as shown in the following table, are used to change object manager parameters.

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

ORACLE

8 Siebel eSales Workflow Reference

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*
- *Activating Siebel eSales Workflows*
- *Siebel eSales Quote and Template Workflows*
- *Siebel eSales Checkout Workflows*
- *Siebel eSales Notification Workflows*
- *Siebel eSales Purchase Approval Workflows*
- *Siebel eSales Order Modification Workflows*
- *Siebel eSales Error Message and View Display Workflows*
- *Auction Workflow*

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, seen *Siebel Business Process Framework: Workflow Guide* .

To modify the error messages that eSales displays, use a subprocess call to the eSales - Display Error Process (see *eSales - Display Error Process*).

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.

ORACLE

Activating Siebel eSales Workflows

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

Siebel eSales uses the following workflows:

- Quotes and Templates

    ○ *eSales - Edit Quote Process*

    ○ *eSales - Order Quote Process*

    ○ *eSales - Reprice Process*

    ○ *eSales - Save Cart Process*

    ○ *eSales - Add To Template Process*

    ○ *eSales - Delete Old Anonymous Quotes*

- Checkout

    ○ *SS Start Checkout Process*

    ○ *Credit Check - Quotes (eSales)*

    ○ *eSales - Enter Shipping Details Process*

    ○ *SS Save Shipping Address Details Process*

    ○ *eSales - Enter Payment Details Process*

    ○ *SS Save Payment Details Process*

    ○ *eSales - Complete Checkout Process*

    ○ *eSales - Authorize Order Process*

    ○ *SS Authorize Order Process*

    ○ *eSales - Save Credit Card*

    ○ *SS Set Shipping Checkout Mode Process*

    ○ *eSales - Save Shipping Address*

    ○ *eSales - Save Account Shipping Address*

    ○ *eSales - Set Primary Account Address*

    ○ *eSales - Setting PO Status*

    ○ *eSales - Current User Type*

    ○ *SS Change Checkout Contact Detail Process*

    ○ *SS Cancel Checkout Process*

- Purchase approval

    ○ *eSales - Approve Order Process*

    ○ *eSales - Reject Order Process*

- Order modification

    ○ *eSales - Cancel Order Process*

    ○ *eSales - Modify Order Process*

    ○ *eSales - Reorder Process*

- Error messages and views

ORACLE

  - *eSales - Display Error Process*
  - *eSales - Login View*
  - *eSales - Order Confirmation View*
  - *SS Order Confirmation View Process*
  - *eSales - Shopping Cart View*

- Auctions

  - *eSales - Silent Authorize Order Process* (included for backward compatibility only)

In addition to the workflows covered in this chapter, Siebel eSales uses *Shipping Calculation Process Workflow*.

- 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*
- *eSales - Order Quote Process*
- *eSales - Reprice Process*
- *eSales - Save Cart Process*
- *eSales - Add To Template Process*
- *eSales - Delete Old Anonymous Quotes*

eSales – Edit Quote Process

The workflow for editing a saved quote is shown in the following image.
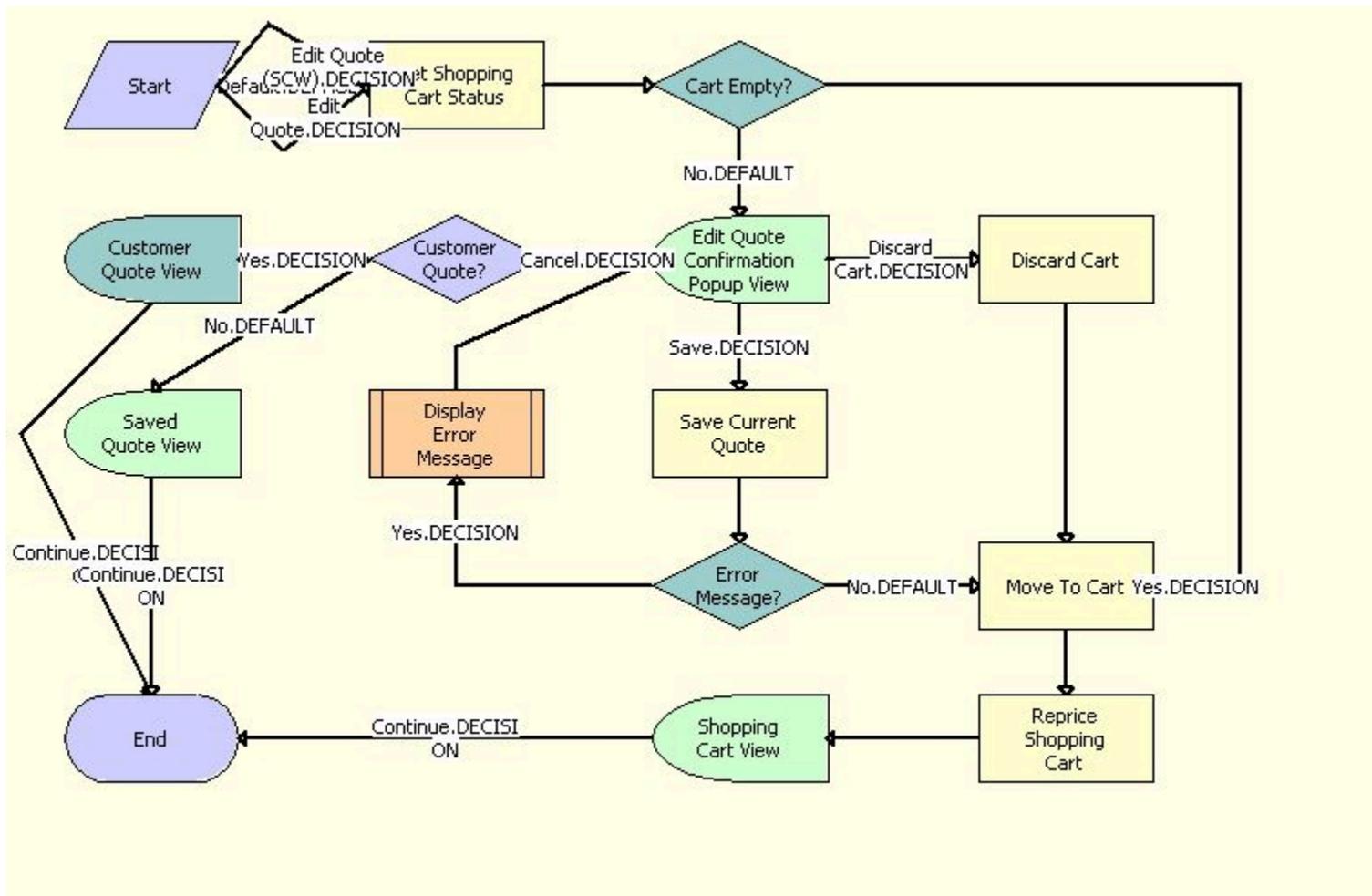
The eSales - Edit Quote Process is called when a user clicks Edit Quote in the Saved Quote Detail view. The following events happen:

- 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:

  o **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.
  o **Discard Cart.** The Shopping Cart is emptied and the quote to be edited is moved to the Shopping Cart.
  o **Save.** If there is an error, the eSales - Display Error Process (see *eSales - Display Error Process*) 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*).

- The Shopping Cart is repriced and displayed.

ORACLE

eSales – Order Quote Process

The workflow for turning saved quotes into orders is shown in the following image.
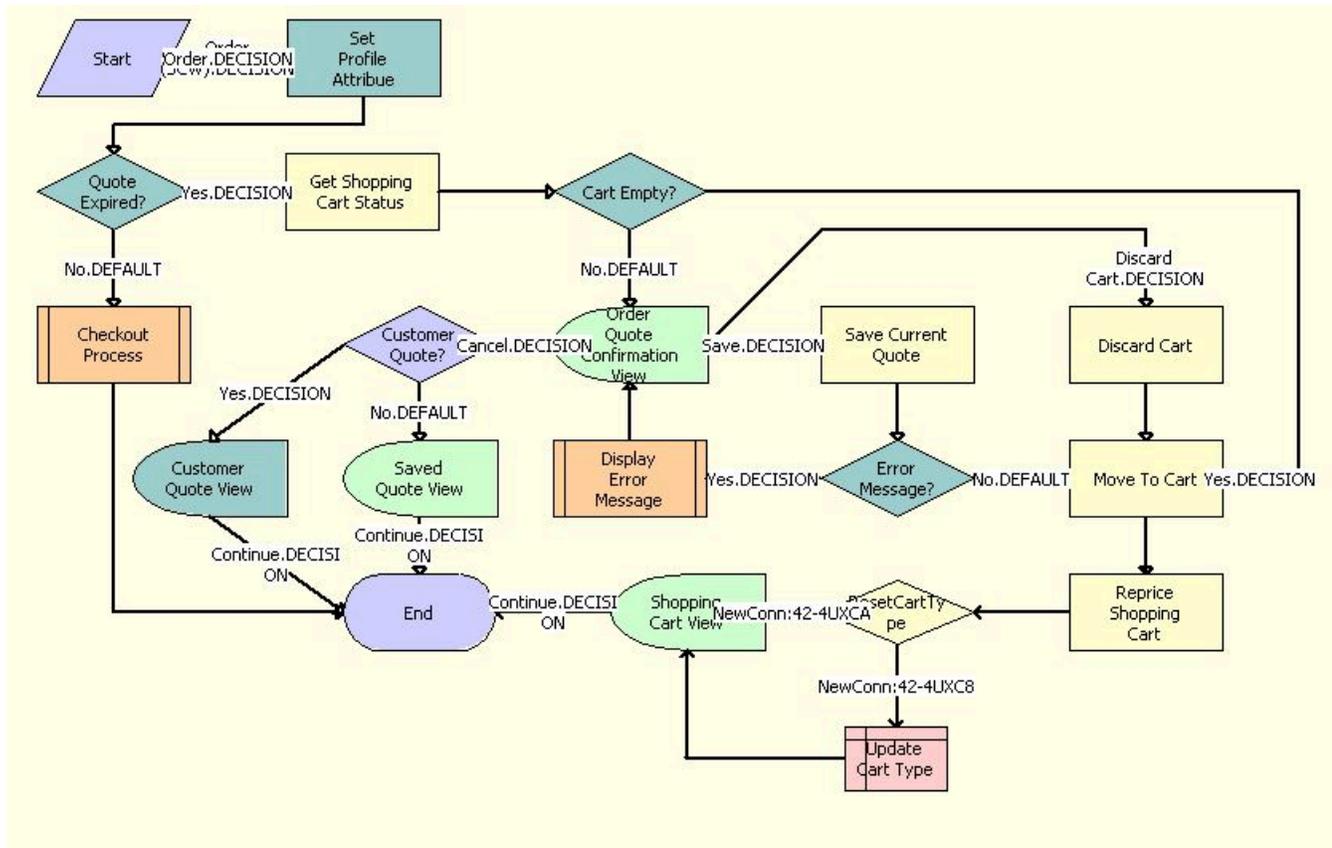
The eSales – Order Quote Process is called when a user orders a saved quote in the Saved Quote Detail view in Siebel eSales. The following events happen:

- The process determines whether the quote has expired. If it has not expired, the SS Start Checkout Process (see *SS Start Checkout Process*) is called.

- If the quote has expired, the process determines whether the Shopping Cart is empty.

  - o If the Shopping Cart is empty, the quote is moved into the Shopping Cart and repriced. The user is sent to the Shopping Cart view.
  - o If the Shopping Cart is not empty, 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 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 *eSales – Display Error Process*) is called, and then the user is sent

ORACLE

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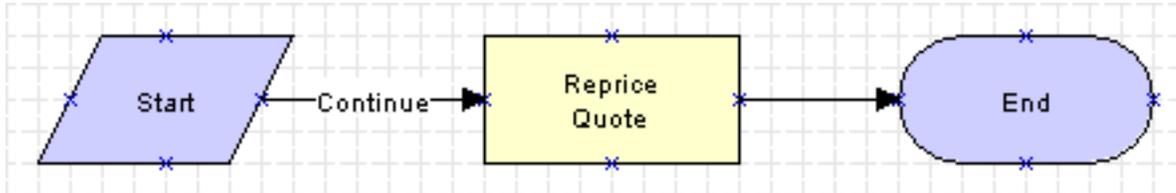
The items in the quote are repriced based upon the current price in the appropriate price list.

The eSales - Reprice Process is called by the eSales - Modify Order Process (see *eSales - Modify Order Process*). 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 the following image.

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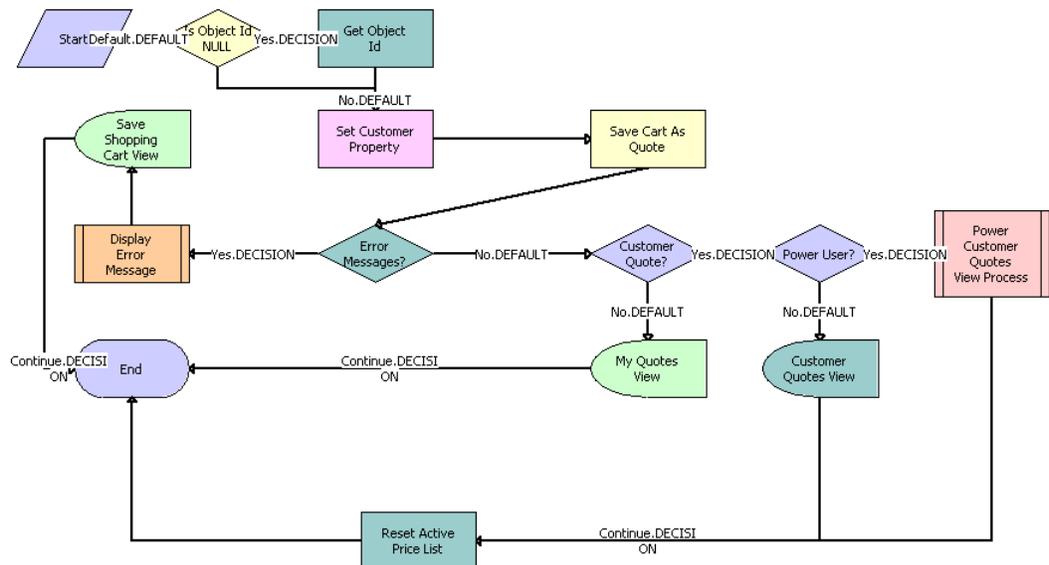
The eSales - Save Cart Process is called when a user clicks Save Cart in the Save Shopping Cart view. The following events happen:

- 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 *eSales - Display Error Process*) 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. If the customer is a power user, Power Customer Quotes View Process is called, followed by Reset Active Price List. If the customer is not a power user, the My Quotes view appears.

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 the following image.

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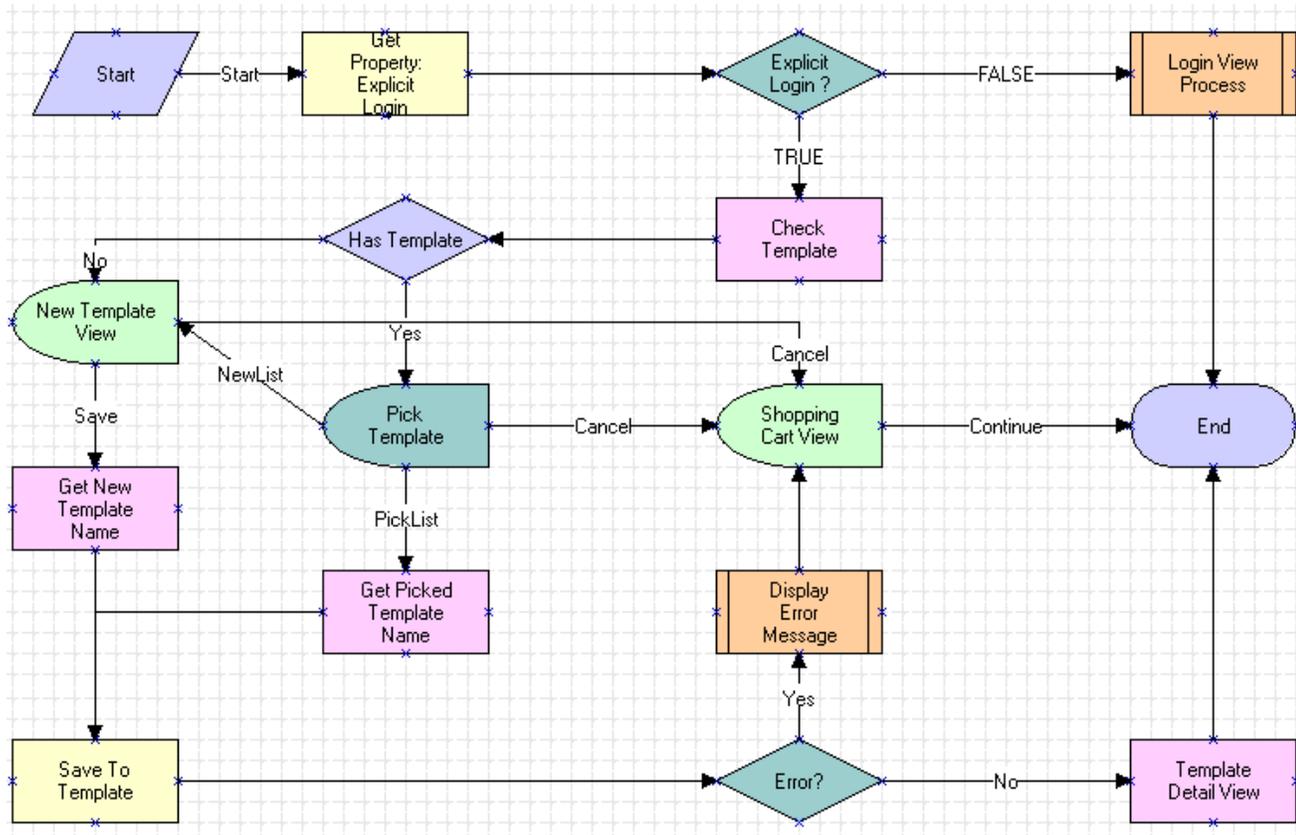
The eSales - Add To Template Process is called when a user clicks the Add To Template button in the shopping cart or catalog product detail view. The following events happen:

- 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 the following image.

ORACLE

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 *Administering Siebel eSales*.

Siebel eSales Checkout Workflows

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

- *SS Start Checkout Process*
- *Credit Check - Quotes (eSales)*
- *eSales - Enter Shipping Details Process*
- *SS Save Shipping Address Details Process*
- *eSales - Enter Payment Details Process*
- *SS Save Payment Details Process*
- *eSales - Complete Checkout Process*
- *eSales - Authorize Order Process*
- *SS Authorize Order Process*
- *eSales - Save Credit Card*
- *SS Set Shipping Checkout Mode Process*
- *eSales - Save Shipping Address*
- *eSales - Save Account Shipping Address*
- *eSales - Set Primary Account Address*
- *eSales - Setting PO Status*
- *eSales - Current User Type*
- *SS Change Checkout Contact Detail Process*
- *SS Cancel Checkout Process*

SS Start Checkout Process

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

ORACLE
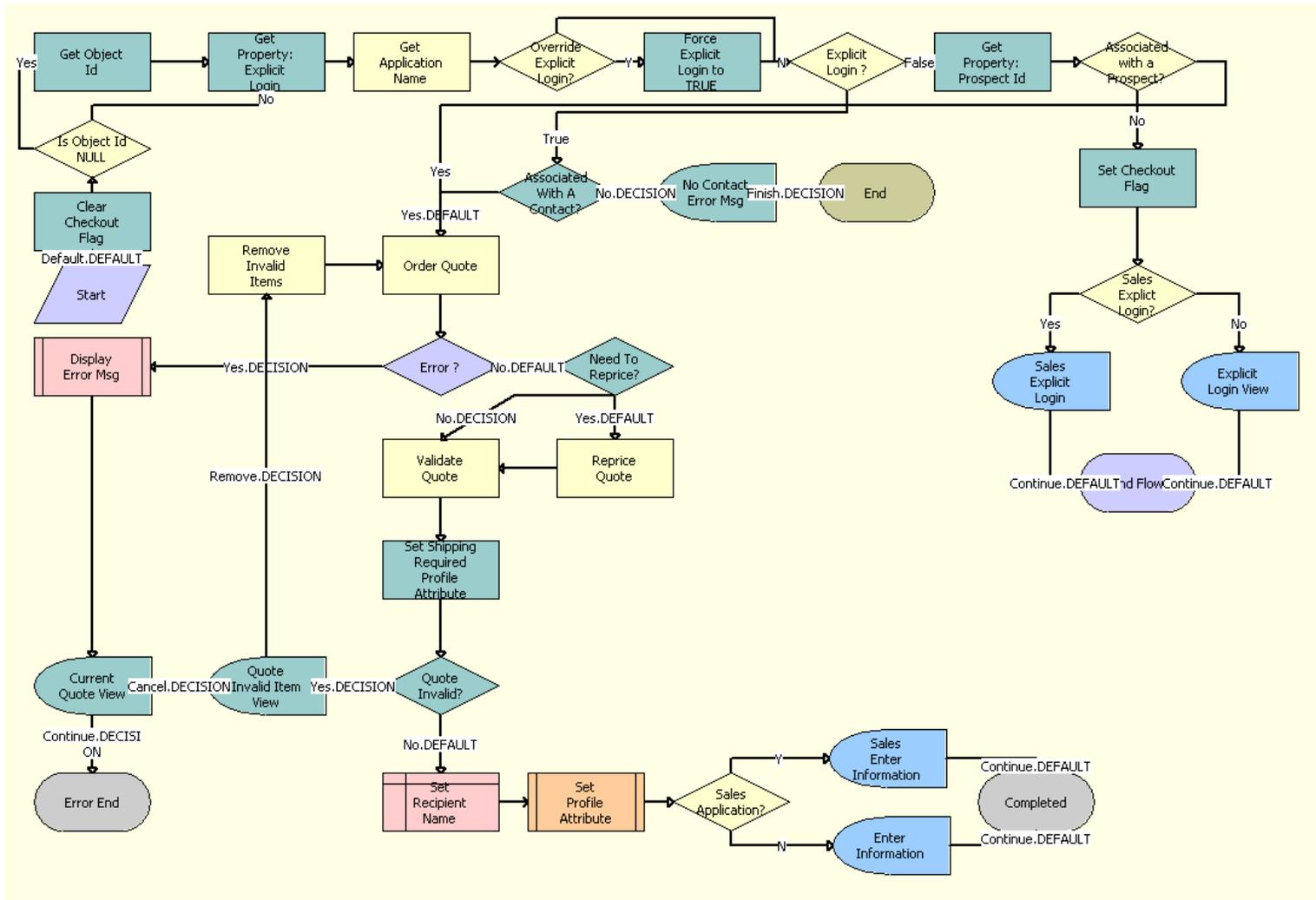
The following events happen:

- 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 the following image.

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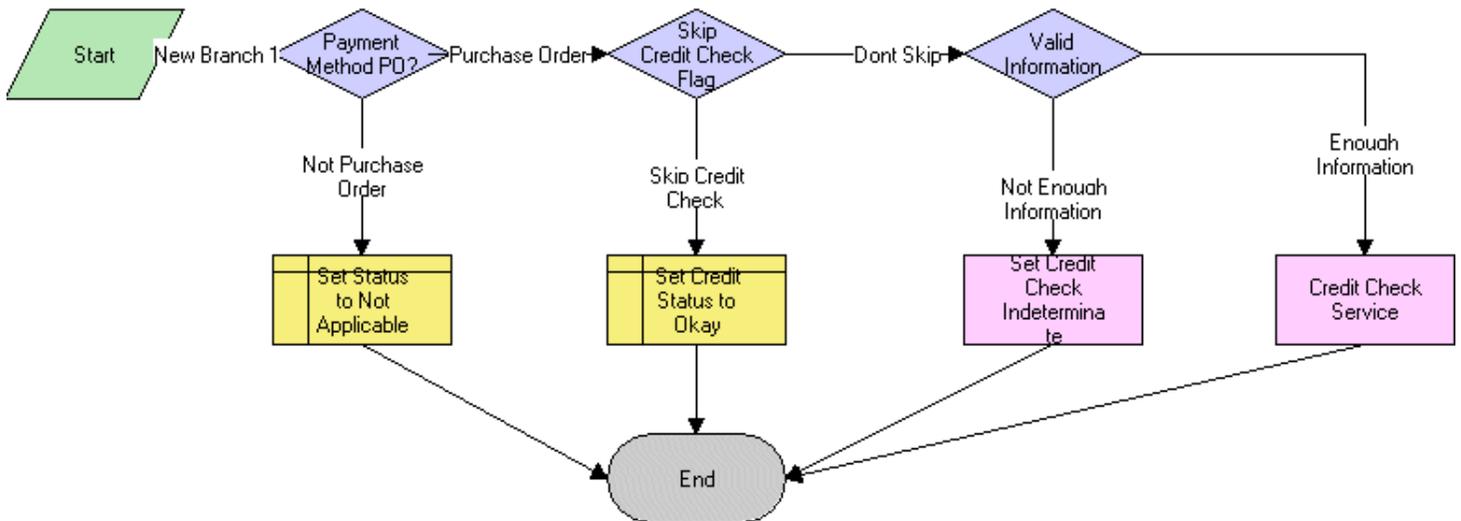
The Credit Check process is called from the eSales – Authorize Order Process. The following events happen:

- 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 less than 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.

eSales – Enter Shipping Details Process

The workflow for selecting, entering, and modifying shipping information is shown in the following image.

ORACLE
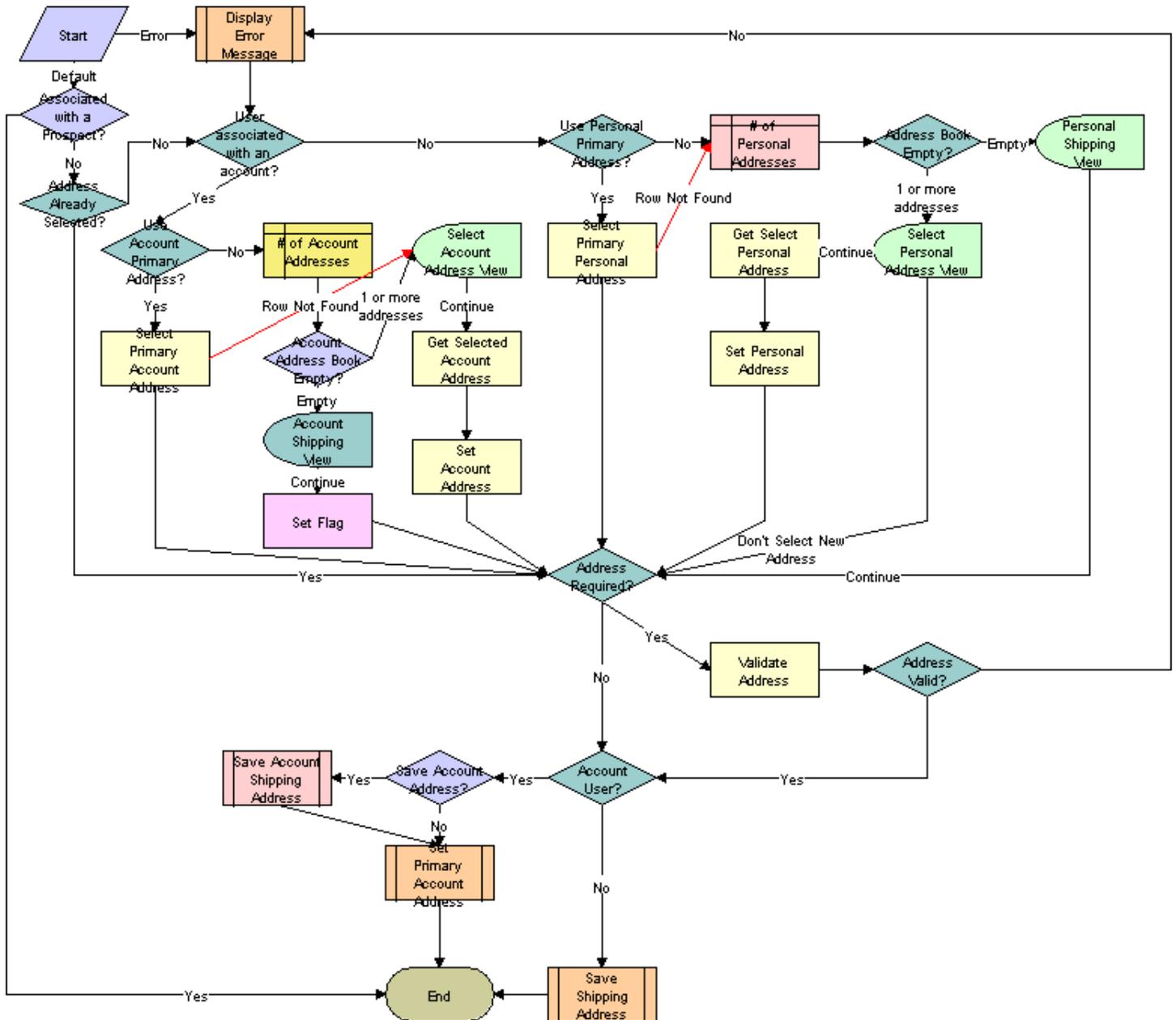
When shipping information needs to be entered or validated, the eSales - Enter Shipping Details Process is called. The following events happen:

- If there is an error in the workflow, the eSales - Display Error Process workflow (see *eSales - Display Error Process*) is called.

- If there is no error, the process determines whether the user has already selected an address. If so, it skips the steps of selecting an address and entering it.

- 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 *eSales - Save Account Shipping Address*), followed by the eSales - Set Primary Account Address subprocess (see *eSales - Set Primary Account Address*). 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 *eSales - Save Shipping Address*) 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 the following image.

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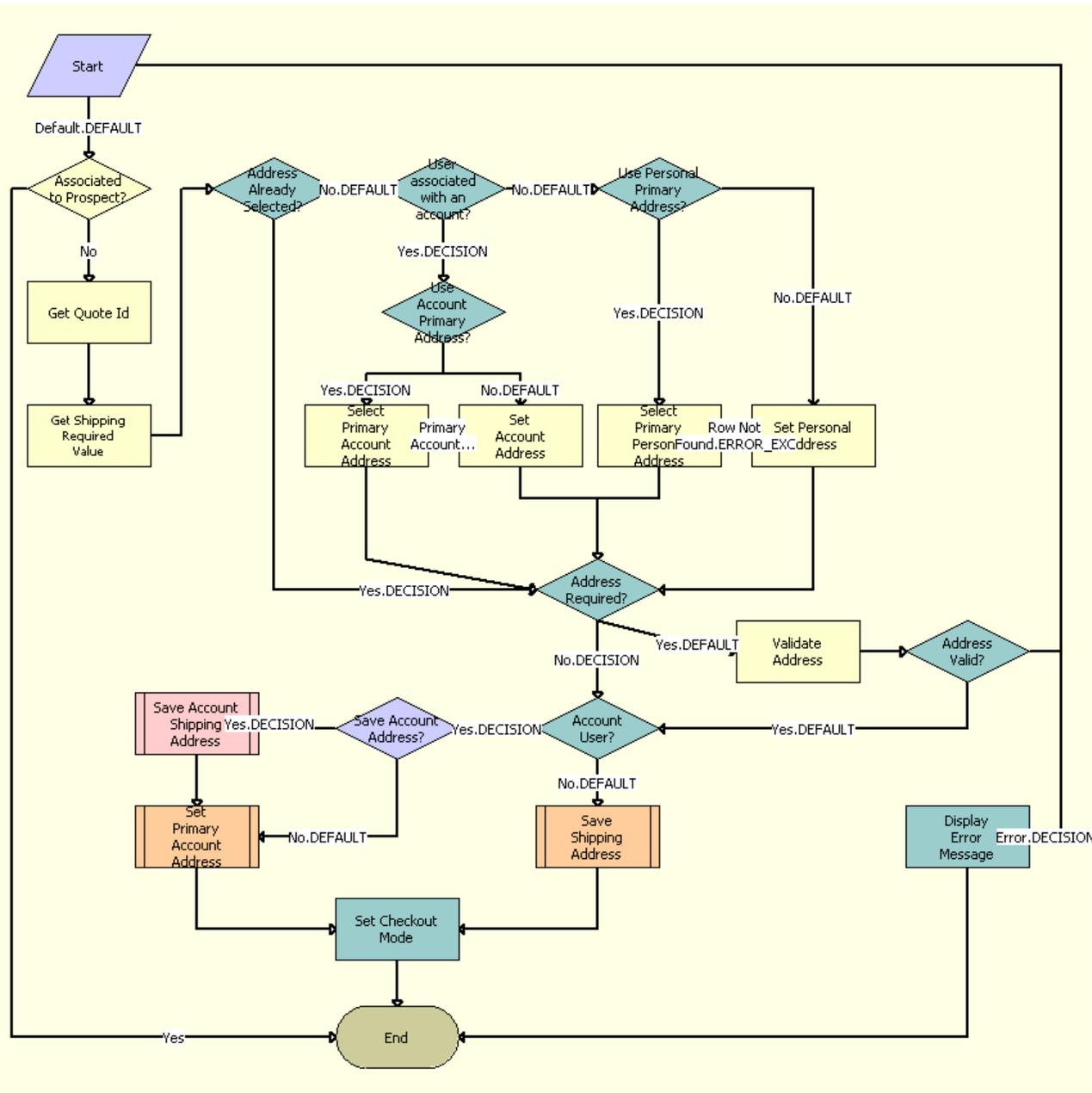
The SS Save Shipping Address Details Process is called when the user selects a Shipping Address and clicks on Continue. The following events happen:

- The process identifies the address selected by the customer and associates it to the quote. If the following 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

ORACLE

- 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 *eSales - Save Account Shipping Address*) to save the address. The process then calls the eSales - Set Primary Account Address subprocess (see *eSales - Set Primary Account Address*) 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 *eSales - Save Shipping Address*) 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 the following image.

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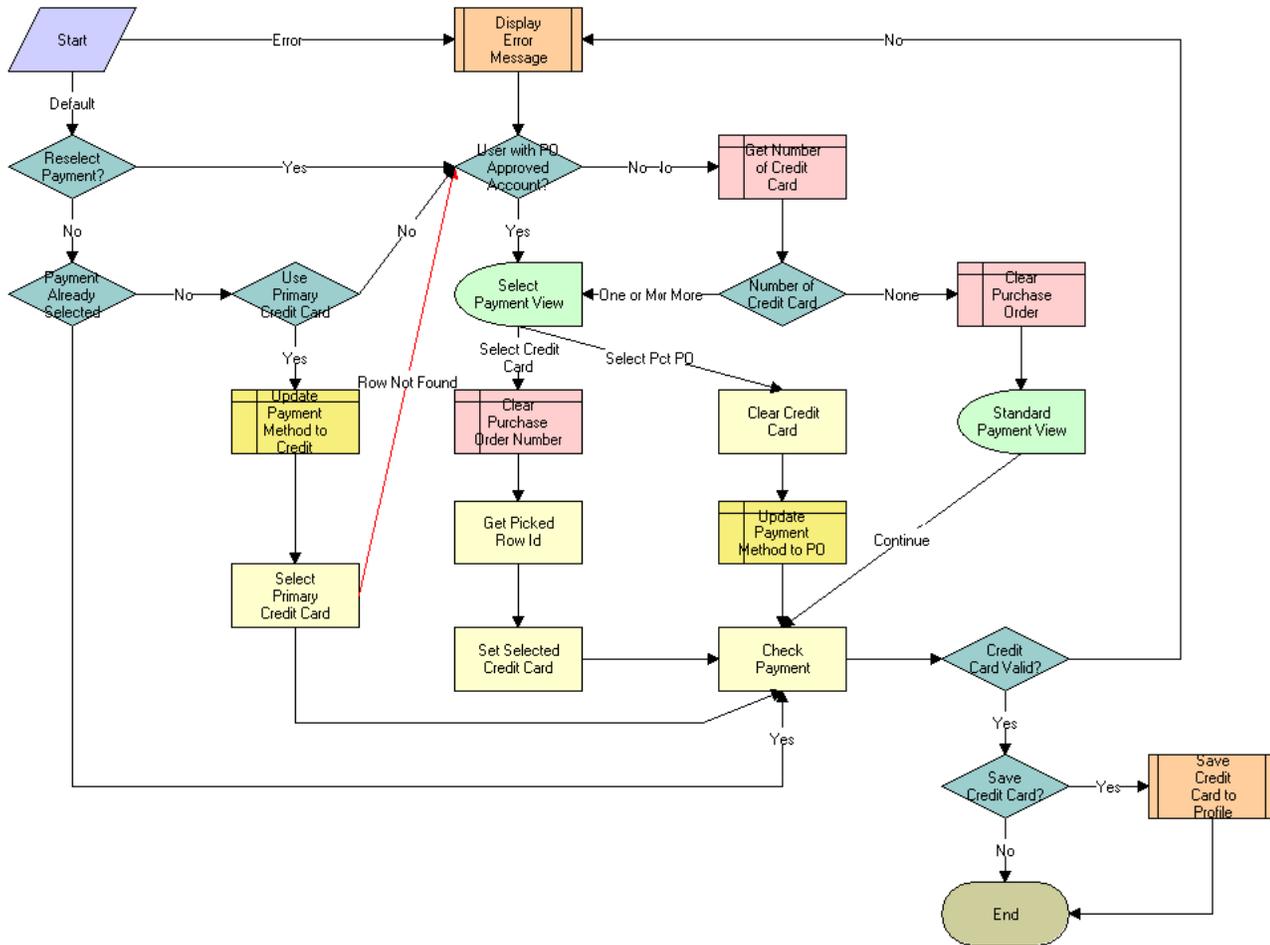
When payment information needs to be entered, the eSales - Enter Payment Details Process is called. The following events happen:

- If there is an error in the workflow, the eSales - Display Error Process (see *eSales - Display Error Process*) 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 the following image.

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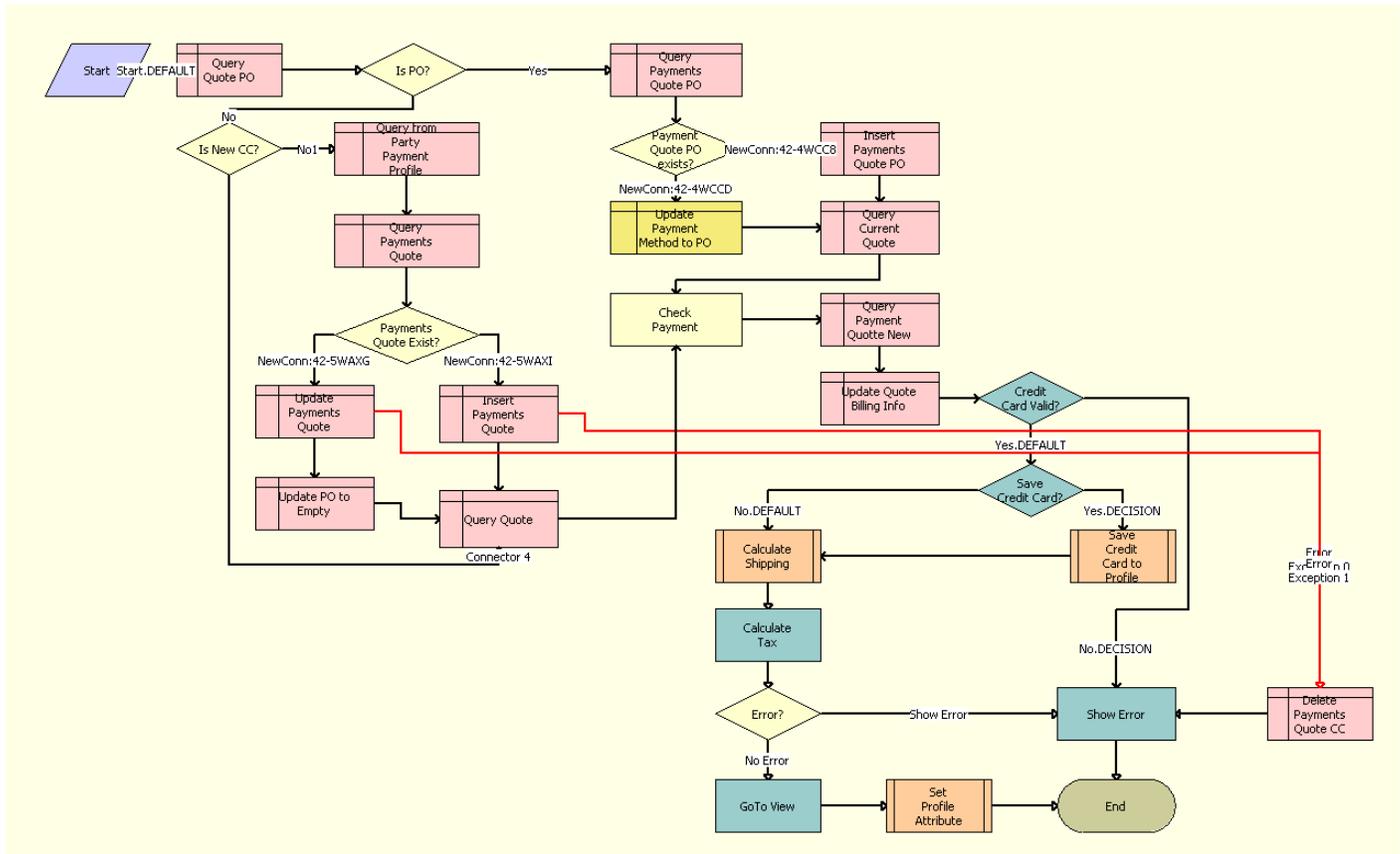
When the user clicks Continue in the Payment applet of the Checkout View, the SS Save Payment Details Process is called. The following events happen:

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

  o The process checks if there is payment information in the profile.

  o If there is payment information available, then the process updates the profile with the current purchase order details

  o 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.

  o 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.

  o 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:

  o Validates the payment .

  o Updates the billing information.

  o Calculates the shipping cost.

ORACLE

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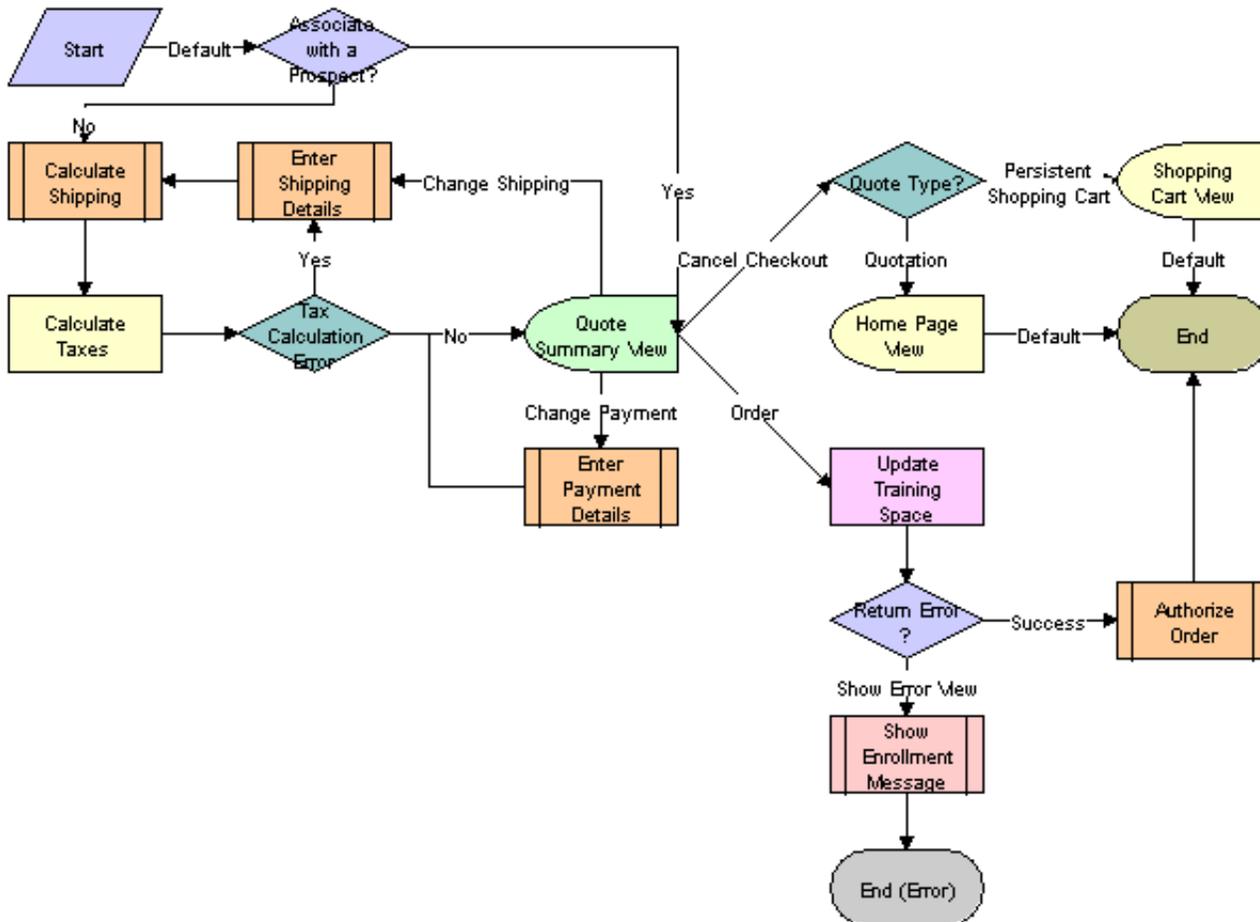
In the eSales - Checkout Process, after shipping and payment information have been added to the order, the eSales - Complete Checkout Process is called. The following events happen:

　　•　Shipping charges are calculated using the Shipping Cost Service (eScript) business service.

ORACLE

- 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*.

- Choosing to confirm the order calls the eSales - Authorize Order Process (see *eSales - Authorize Order Process*).

eSales – Authorize Order Process

The workflow for authorizing credit card payments and purchase orders is shown in the following image.

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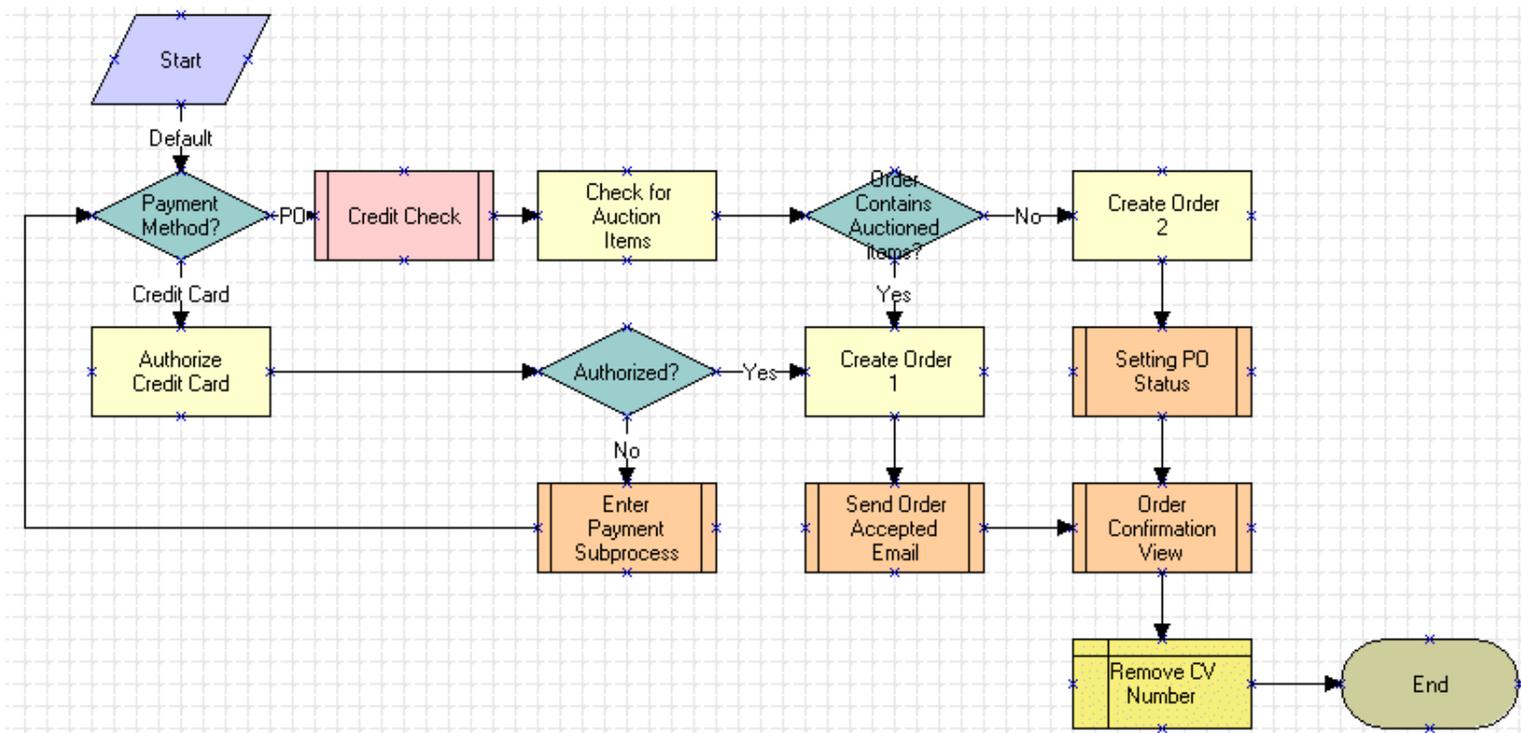
Confirming the order in the Order Summary view calls the eSales - Authorize Order Process. The following events happen:

- The process determines which payment method was selected, PO or credit card.

- If the PO method was selected, the process:

  o 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.

  o 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 *eSales - Setting PO Status*), and then the Order Confirmation view is shown using the eSales - Order Confirmation View process (see *eSales - Order Confirmation View*).

    - 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.

  o 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.

  o If authorization fails, the eSales - Enter Payment Details Process (see *eSales - Enter Payment Details Process*) is called. Credit card authorization is performed again and if it is successful, the order is created.

See *Using Other 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 the following image.

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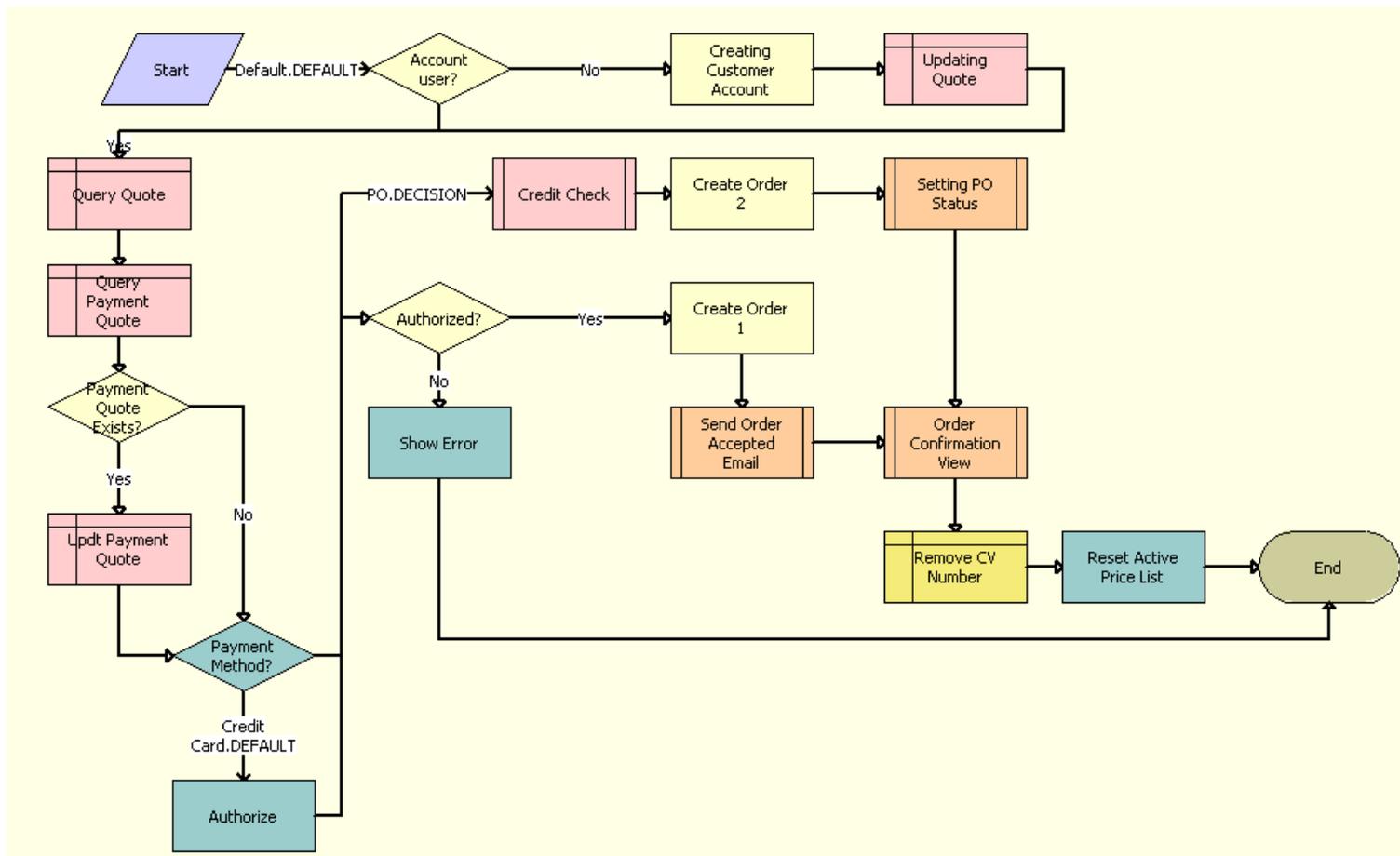
This workflow also performs a credit card authorization where a credit card is used, and where a credit check is required in a purchase order. The following events happen:

- 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*), which sends an order acceptance email to the customer.

- The process calls the SS Order Confirmation View Process (see *SS Order Confirmation View Process*) to navigate to the Order Confirmation view.

- For purchase orders, the process calls the eSales - Credit Check - Quotes (see *Credit Check - Quotes (eSales)*) to perform a credit check.

- For purchase order flows, a sales order is created for the current quote.

ORACLE

- For purchase orders, the process calls the eSales - Setting PO Status workflow (see *eSales - Setting PO Status*) 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 the following figure.
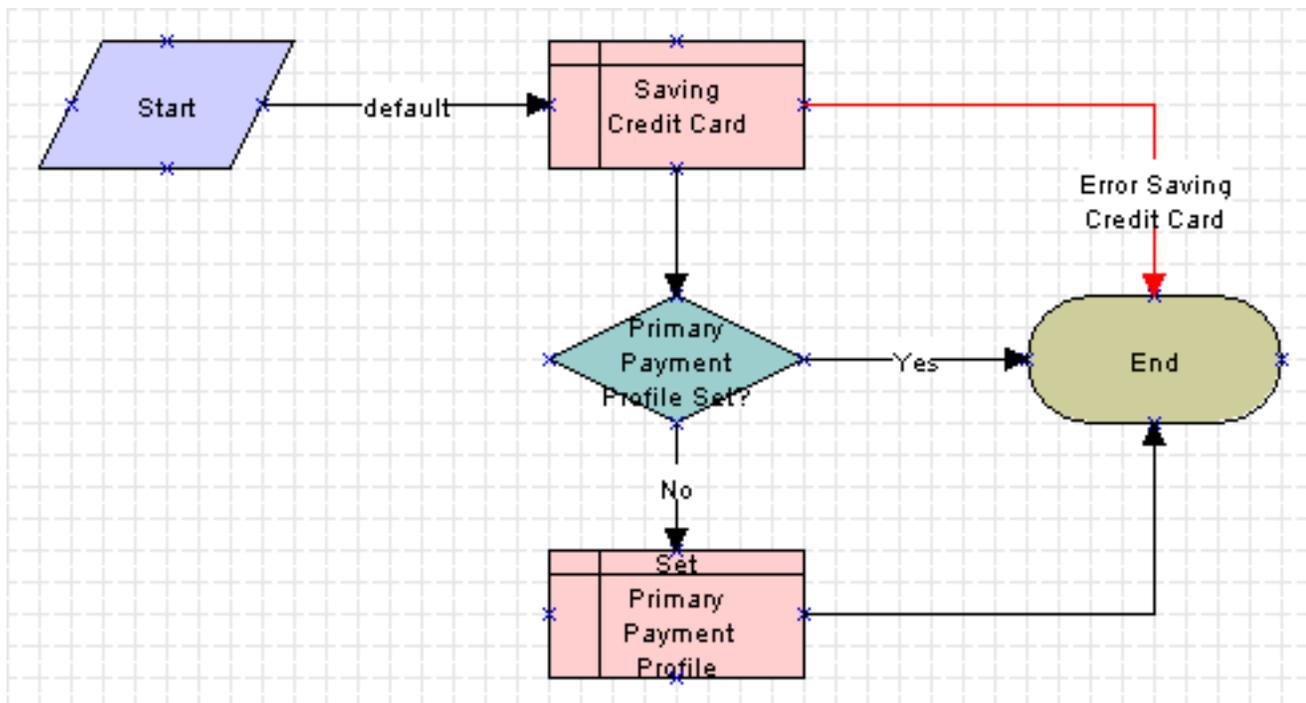
When the eSales - Save Credit Card process is called by the eSales - Enter Payment Details Process (see *eSales - Enter Payment Details Process*), the following events happen:

- 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 the following image.

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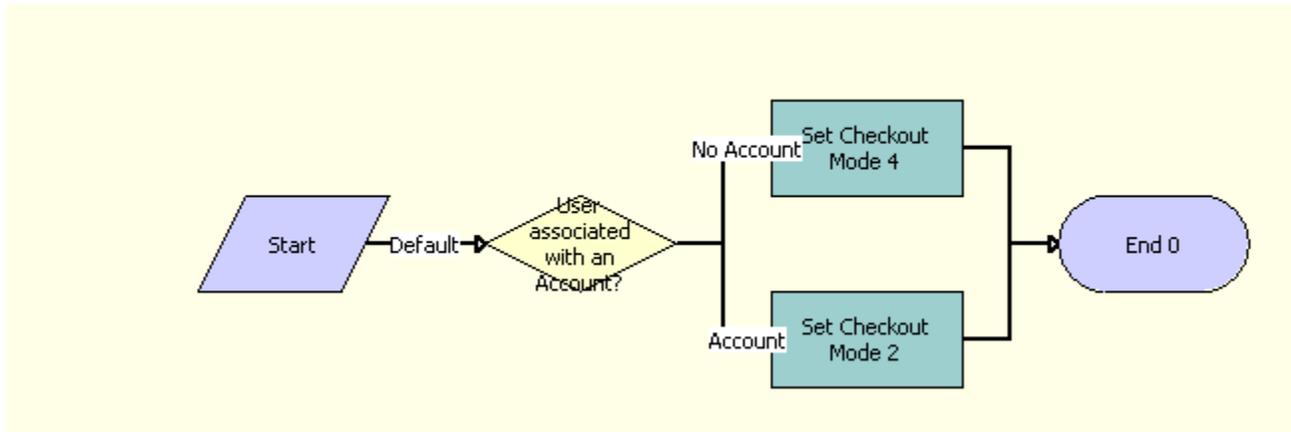
The SS Set Shipping Checkout Mode Process is called when the user performs one of the following actions:

- 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 the following figure.

ORACLE
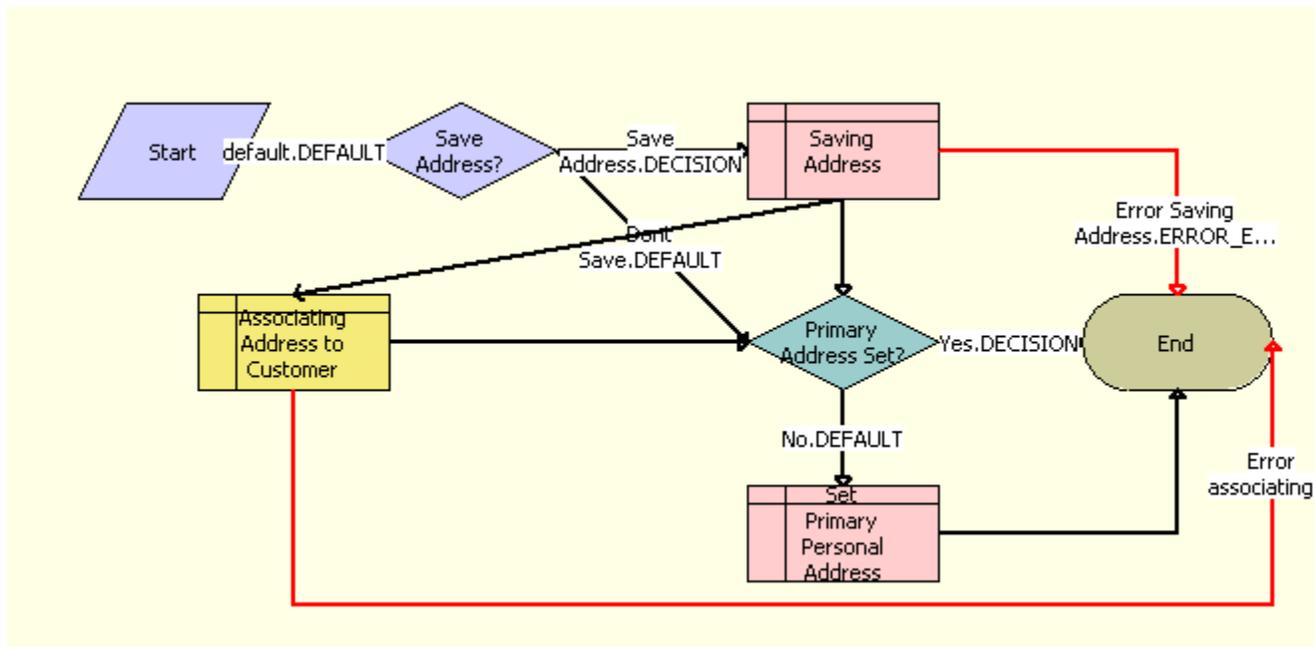
When the eSales - Save Shipping Address workflow is called by the SS Save Shipping Details Process (see *SS Save Shipping Address Details Process*), the following events happen:

- 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 the following figure.
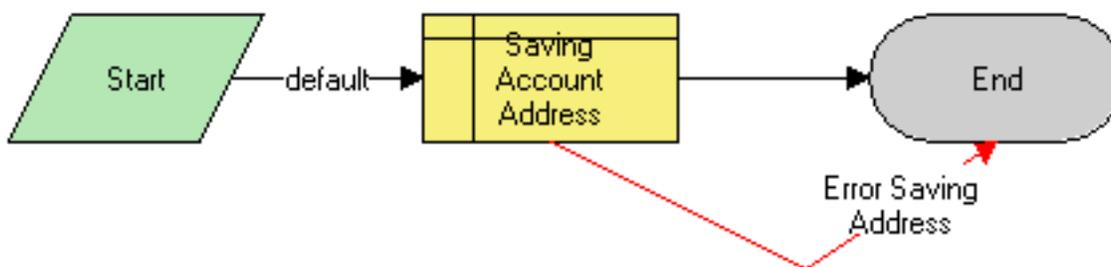
When the eSales - Save Account Shipping Address workflow is called by the SS Save Shipping Details Process (see *SS Save Shipping Address Details Process*), it saves the account shipping address to the CUT Address business component. If there is an error saving the address, the workflow ends.

ORACLE

eSales – Set Primary Account Address

The workflow for setting the primary account address is shown in the following figure.
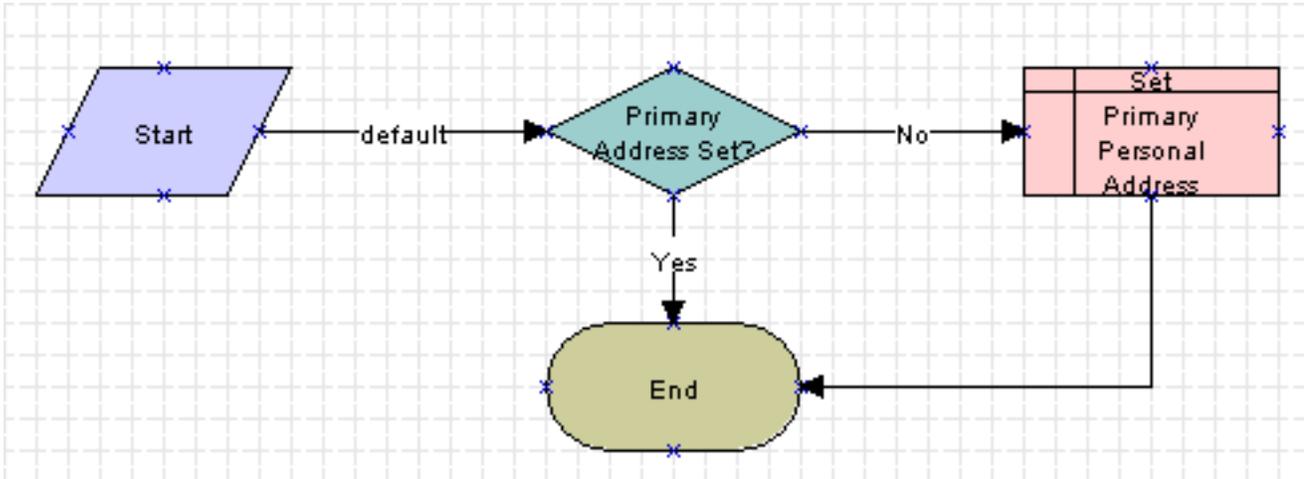
When the eSales - Set Primary Account Address workflow is called by the SS Save Shipping Details Process (see *SS Save Shipping Address Details Process*), it determines whether the B2B user has a primary account address.

- 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 the following image.

ORACLE
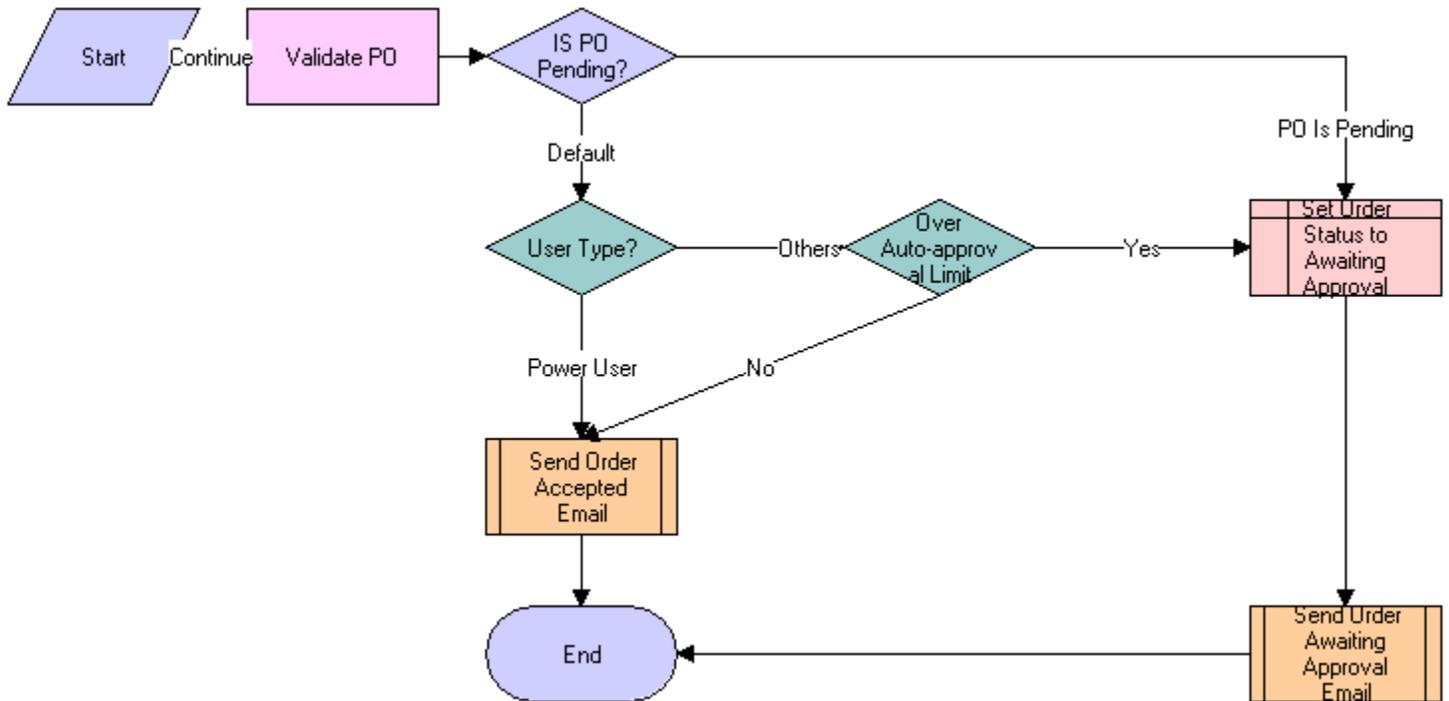
When the eSales - Setting PO Status workflow is called by the eSales - Authorize Order Process (see *eSales - Authorize Order Process*), the following events happen:

- 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*) 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 *Administering Siebel eSales*

eSales - Current User Type

The workflow for looking up the current user type from the contact record is shown in the following image.

ORACLE
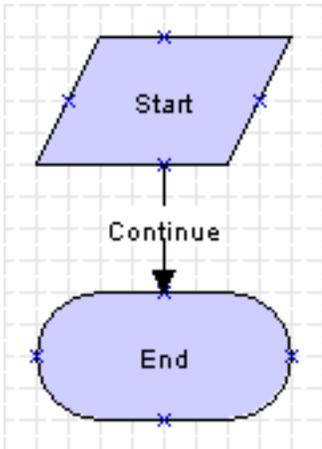
The eSales - Current User Type workflow is called by the eSales - Approve Order Process (see *eSales - Approve Order Process*) and the eSales - Reject Order Process (see *eSales - Reject Order Process*).

SS Change Checkout Contact Detail Process

The workflow for changing the contact details during checkout is shown in the following image.
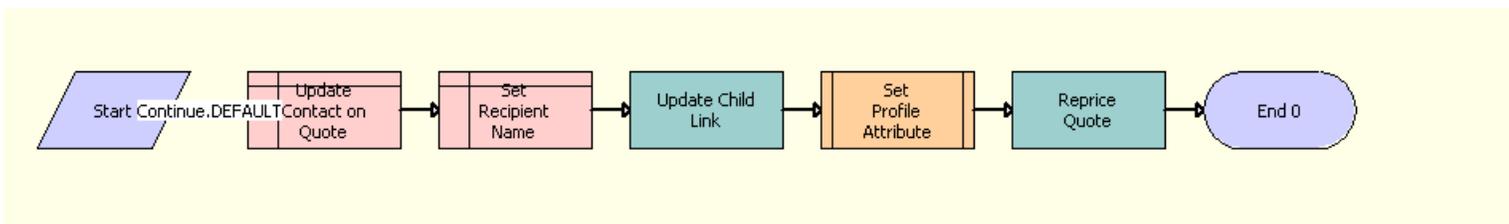
When the SS Change Checkout Contact Detail workflow is called, the following happens:

- 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 *SS Set Shipping Checkout Mode Process*) 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 the following image.

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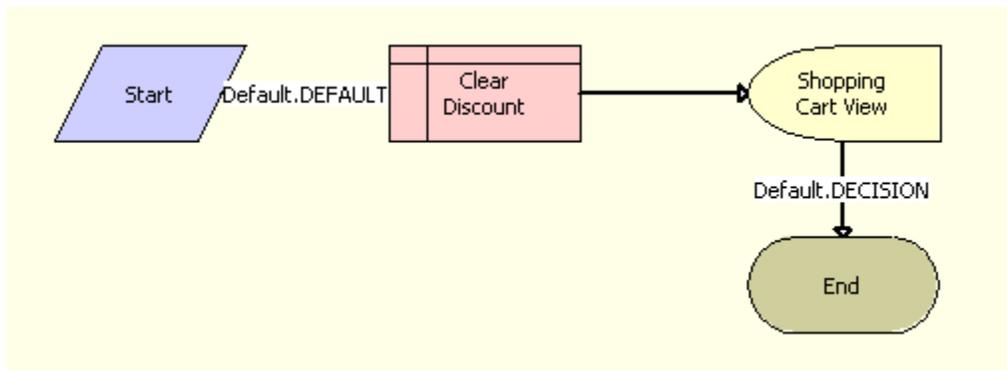
When the SS Cancel Checkout Detail Process workflow is called, the process clears any discount applied to the quote and returns to the Shopping Cart view.

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 the following image. The process sends an email or an error generates an exception.
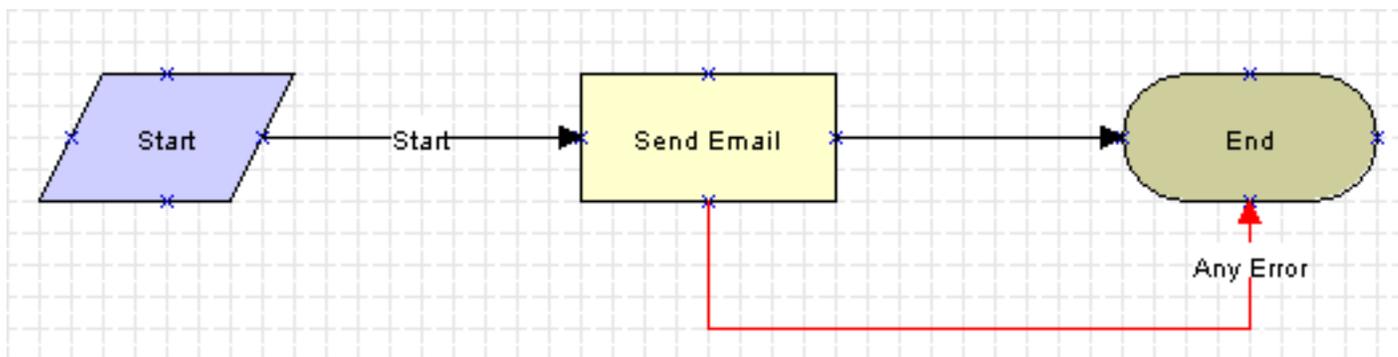
For more information on email notifications, see *Customizing Email Notifications for Siebel eSales*. For information on workflow policies and workflow policy actions, see *Siebel Business Process Framework: Workflow Guide* .

ORACLE

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*
- *eSales - Reject Order Process*

eSales - Approve Order Process

The workflow for approving purchase orders is shown in the following image.
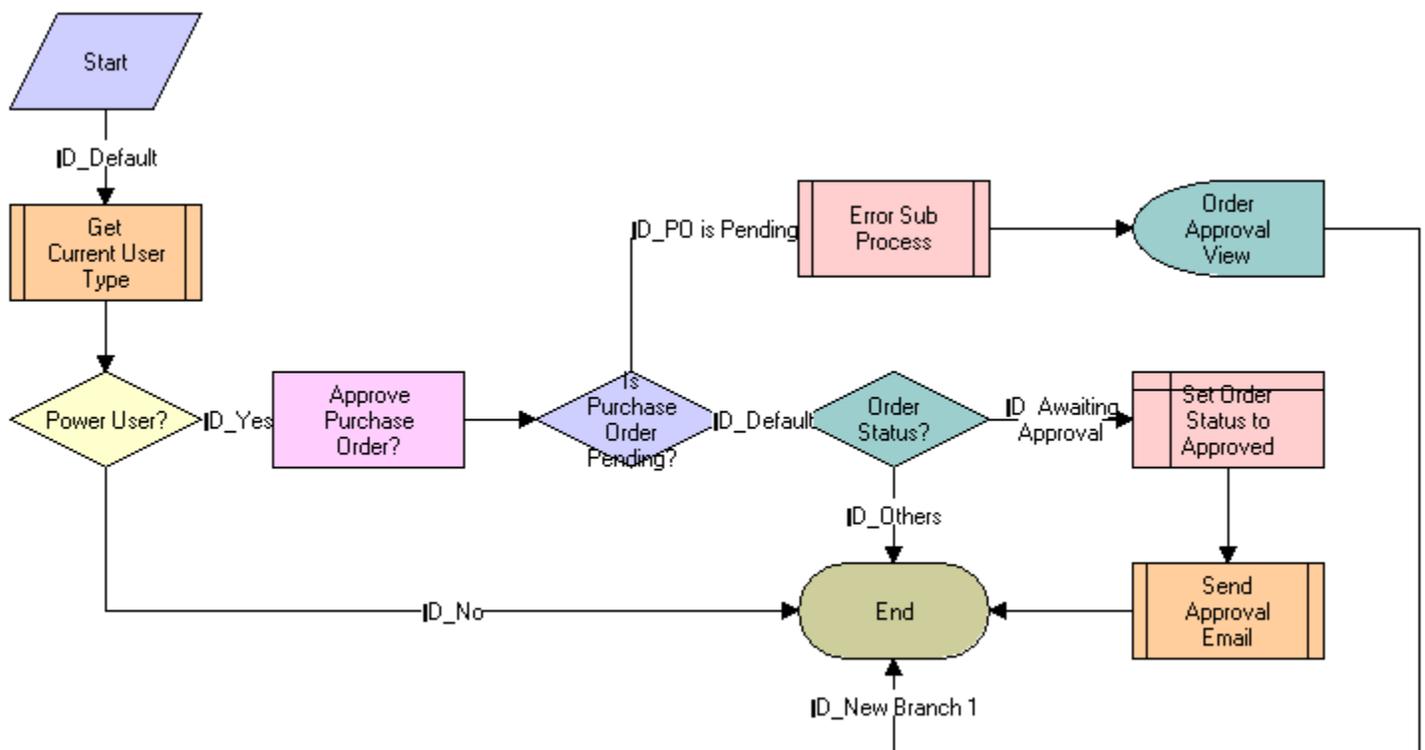
The eSales - Approve Order Process is called when a delegated customer administrator or purchasing manager clicks Approve in the My Orders view. The following events happen:

- The process calls the eSales - Current User Type workflow (see *eSales - Current User Type*) 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

ORACLE

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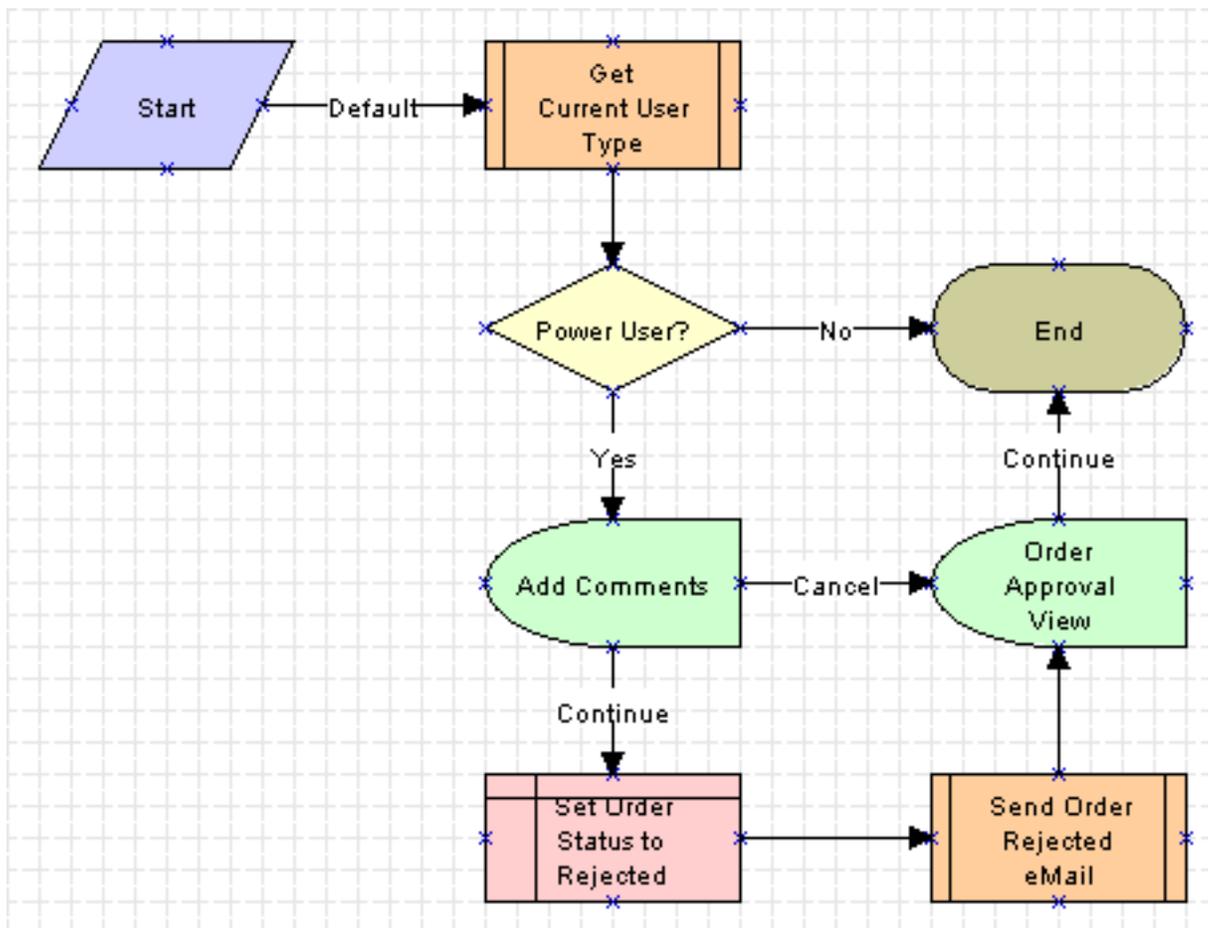
The eSales – Reject Order Process is called when a delegated customer administrator or purchasing manager clicks Reject in the My Orders view. The following events happen:

- The process calls the eSales – Current User Type workflow (see *eSales – Current User Type*) to determine the user type.

- If the user is not a power user, the process ends.

ORACLE

- 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*) 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*
- *eSales - Modify Order Process*
- *eSales - Reorder Process*

eSales – Cancel Order Process

The workflow for canceling orders is shown in the following image.

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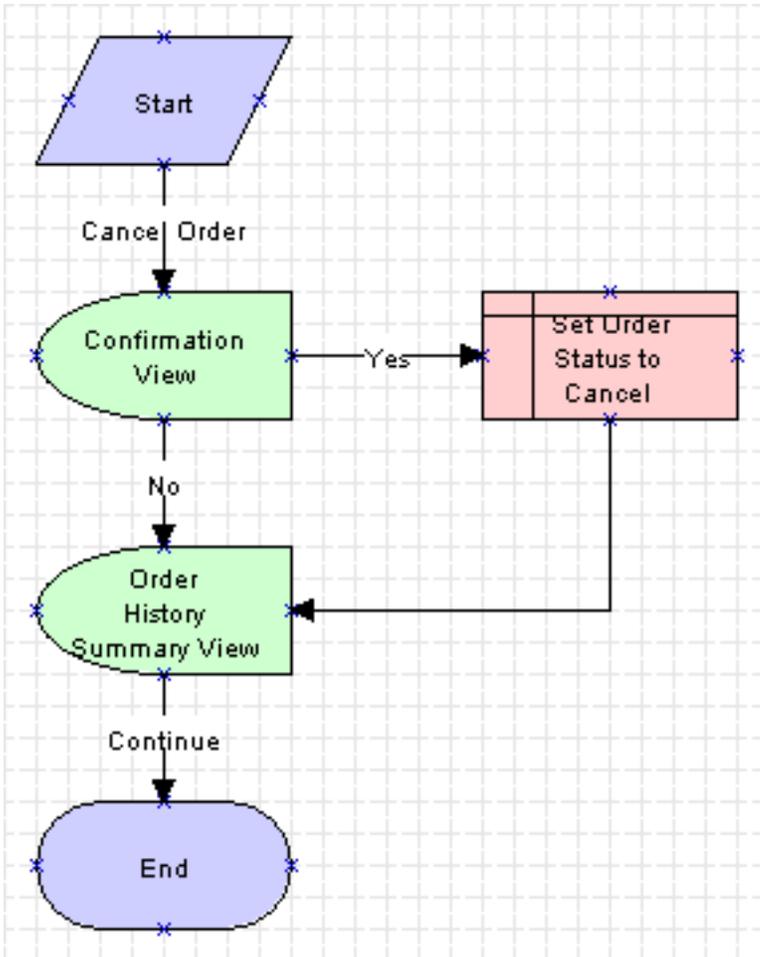
The eSales - Cancel Order Process is called when a customer clicks Cancel in the Order History Summary view. The following events happen:

- 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 the following image. 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.

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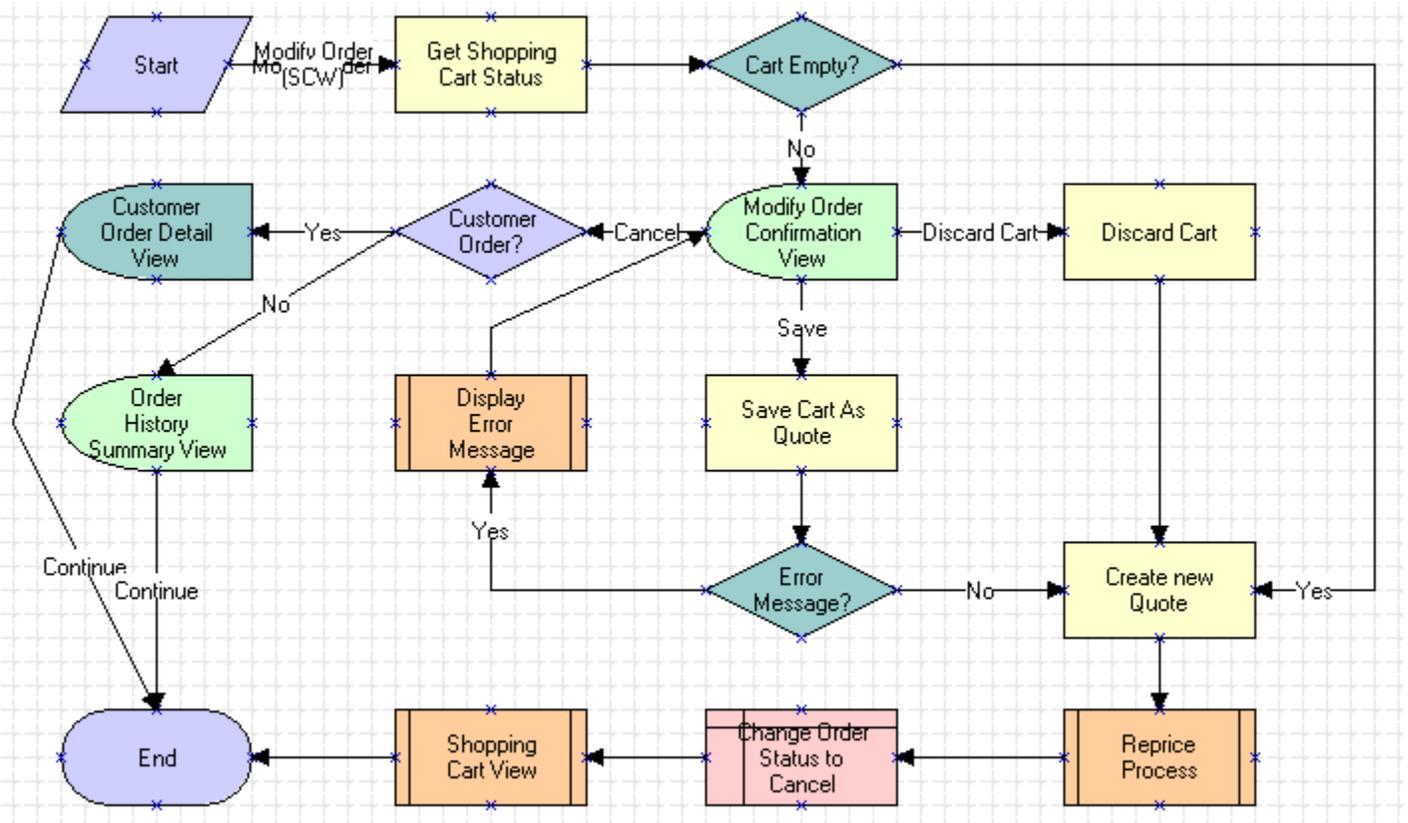
The eSales - Modify Order Process is called when a customer clicks Modify in the Order History Summary view. The following events happen:

- 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 *eSales - Shopping Cart View*) is used to display the Shopping Cart.

  ○ **Save Cart.** If there is an error, the eSales - Display Error Process (see *eSales - Display Error Process*) 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*) 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.

ORACLE

eSales - Reorder Process

The workflow for reordering is shown in the following image. It works similarly to modifying an order.
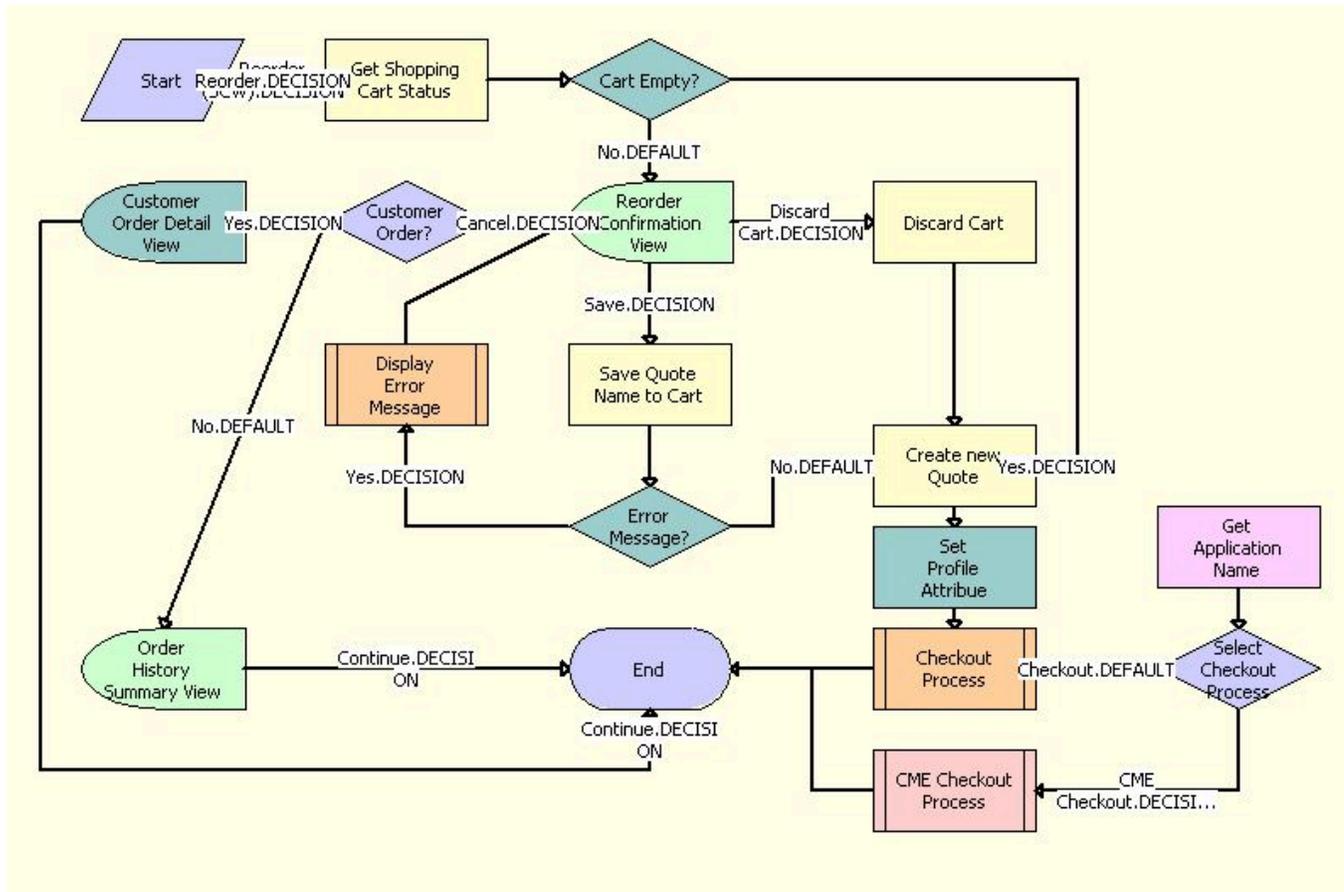
The eSales - Reorder Process is called when a customer clicks Reorder in the Order History Summary view. The following events happen:

- 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 *SS Start Checkout Process*) 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.

ORACLE

- **Save Cart.** If there is an error, the eSales - Display Error Process (see *eSales - Display Error Process*) 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*) 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*
- *eSales - Login View*
- *eSales - Order Confirmation View*
- *SS Order Confirmation View Process*
- *eSales - Shopping Cart View*

eSales - Display Error Process

The process used for displaying error messages is shown in *eSales - Display Error Process*. It is called by the following workflows:

- eSales - Edit Quote Process (*eSales - Edit Quote Process*)
- eSales - Save Cart Process (*eSales - Save Cart Process*)
- eSales - Enter Shipping Details Process (*eSales - Enter Shipping Details Process*)
- eSales - Checkout Process (*SS Start Checkout Process*)
- eSales - Enter Payment Details Process (*eSales - Enter Payment Details Process*)
- eSales - Modify Order Process (*eSales - Modify Order Process*)
- eSales - Reorder Process (*eSales - Reorder Process*)

The following image describes the eSales - Display Error Process workflow.
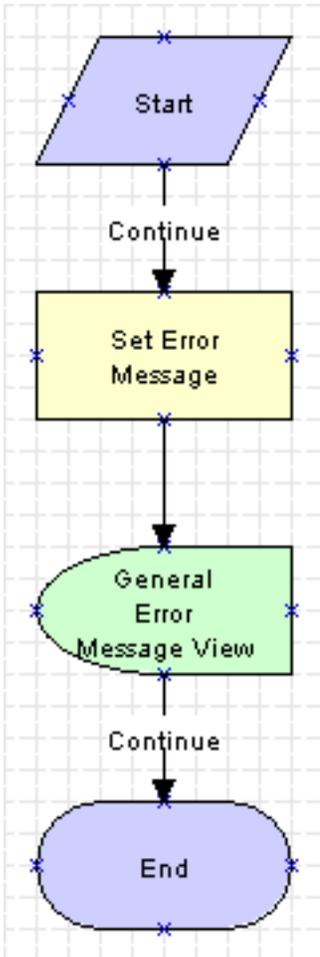
When the eSales - Display Error Process is called, the following events happen:

- 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 the following image.

ORACLE
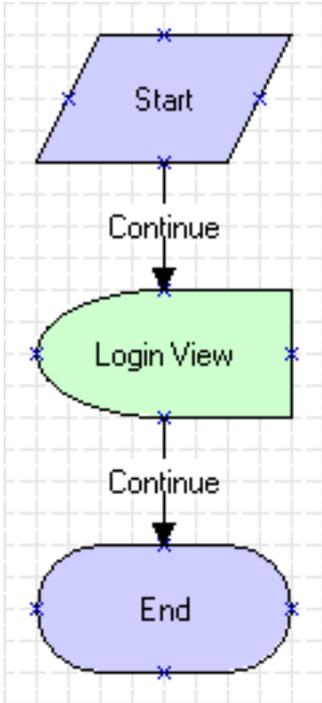
This process is called by the eSales - Checkout Process (see *SS Start Checkout Process*) when an anonymous user tries to check out. The eSales - Login View process is necessary because the Login view is based on the Login business object, while the eSales - Checkout Process is based on the Catalog business object.

eSales – Order Confirmation View

The process for displaying the Order Confirmation view is shown in the following image.

ORACLE
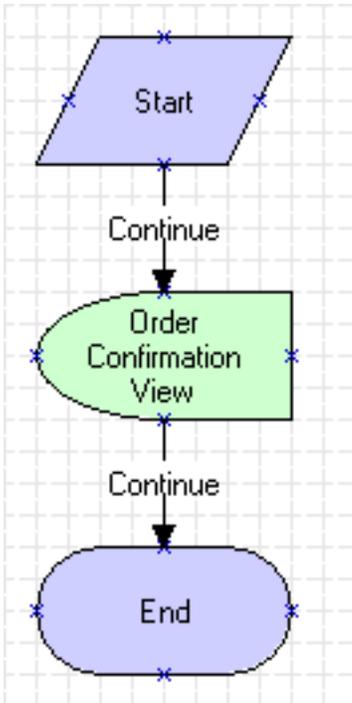
This process is called by the eSales - Authorize Order Process (see *eSales - Authorize Order Process*). The eSales -
Order Confirmation View process is necessary because the Order Confirmation view is based on the Order Entry (Sales)
business object, while the eSales - Authorize Order Process is based on the Catalog business object.

SS Order Confirmation View Process

The workflow for displaying the Order Confirmation view is shown in the following figure.
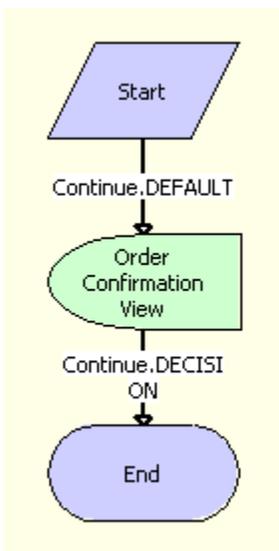
This process is called by the SS - Authorize Order Process (see *SS Authorize Order Process*). The eSales - Order
Confirmation View process navigates the user to the Order Confirmation View.

ORACLE

eSales – Shopping Cart View

The process for displaying the Shopping Cart is shown in the following image.
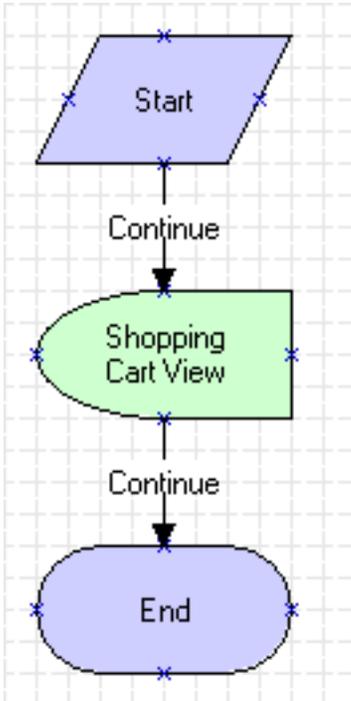
This process is called by the eSales - Modify Order Process (see *eSales - Modify Order Process*). The eSales - Shopping Cart View process is necessary because the Shopping Cart is based on the Catalog business object, while the eSales - Modify Order Process is based on the Order Entry (Sales) business 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*. For information on workflow policies, see *Siebel Business Process Framework: Workflow Guide* .

eSales – Silent Authorize Order Process

The eSales - Silent Authorize Order Process, shown in the following image, is triggered by the Silent Authorize Order workflow policy action.

ORACLE

Note: Auctions are not part of the current Siebel Business Applications functionality. This workflow is included for backward compatibility only.
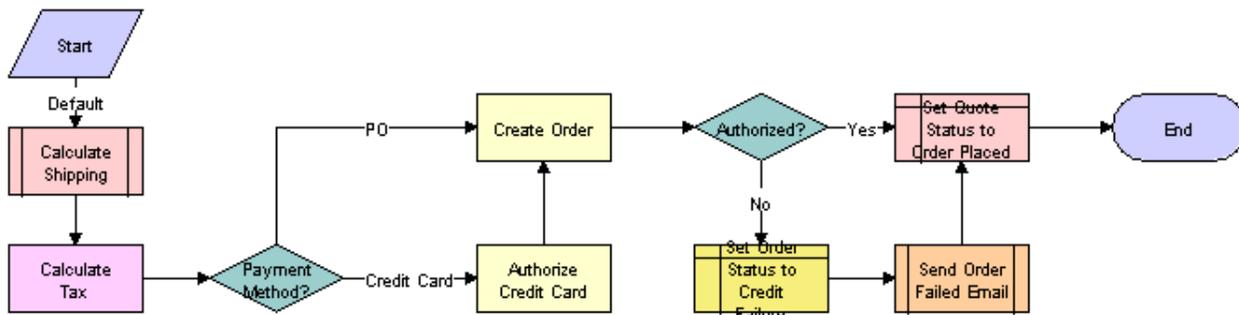
When this workflow is triggered, the following events happen:

- The Shipping Calculation Process is called. For more information, see *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.

9 **Shopping Service Methods Reference**

Shopping Service Methods Reference

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

- *About the Shopping Service Business Service*
- *Siebel eSales User Action Methods*
- *Siebel eSales Workflow Methods*

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*
- *AddToCart Method*
- *AddToFavoriteList Method*
- *SetShoppingCart Method*
- *SetApplicationName Method*
- *OrderTemplate Method*
- *SetShoppingCartInfoChanged Method*
- *GetCartInfo Method*

AddItem Method

This method is similar to *AddToCart Method*.

ORACLE

The only difference is that AddItem can be used to AddItems to the shopping cart of 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 the Action field, the Shopping Cart takes different actions, as described in the following table.

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.

ORACLE

- o If it is Quotation, then the current quote is a normal quote.

- o 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.

- o 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 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 the following table.

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. 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. 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.

ORACLE

Name	Value	Explanation
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.
SetPriceListSearch	Y	Indicates that the `[Price List Id] = GetProfileAttr("PriceListId")` search specification should be applied to the business component. If used, only products that are on the current price list are visible in Siebel eSales.
Use Promotional Price	Y	Used in Attribute Adjustments. 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 (See the following table.)

- **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 (See the following table.)

- **Server component parameters.** The server component parameters used by the Shopping Service business service are listed in the following table. You can change these parameters by navigating to Server Administration > Components > Component Parameters.

Name	Default Value	Explanation
AddToCartAutoQuote	TRUE	The value can be TRUE or FALSE.

ORACLE

Name	Default Value	Explanation
		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. 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. If it is TRUE, AddToCart allows anonymous users to add items to the Shopping Cart, and it creates quotes with quote type Anonymous Quote. 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. 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. If it is TRUE, the application supports the Persistent Shopping Cart. The quote created by AddToCart will have type Persistent Cart. When the user logs in to the application, the run-time 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 the following table, using the Catalog business object as an example.

Business Component	Link	Comment
Internal Product by Price List Optional	None	Underlying business component for AddToCart, QuickAdd, and Customize. All other business components by default use this business component as the AddToCart Proxy Buscomp.

ORACLE

Business Component	Link	Comment
Internal Product by Price List Optional 2	Catalog Category/Internal Product by Price List Optional 2	Used in the Product list applet for AddToCart. 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	None	Needed for extended attributes.
Product XA VBC Form	Internal Product by Price List Optional/Product XA VBC Form	Needed for extended attributes.
Quote	None	None
Quote Item	Quote/Quote Item	None
Quote Item XA	None	None
Training Class Registration	None	Needed for AddTrainingItem.
eEvents Event Attendee	None	Needed for AddTrainingItem.
eEvents Session Attendee	None	Needed for AddTrainingItem.
Favorite List	None	Needed for AddToFavoriteList.
Favorite List Item	Favorite List/Favorite List Item	Needed for AddToFavoriteList.

Arguments for the Add To Cart Method

The arguments for the AddToCart method are shown in the following table.

Name	Type	Data Type	Optional	Comments
Action	Input	String	Y	Specifies the action to take.

ORACLE

Name	Type	Data Type	Optional	Comments
				Action can take the following values: - AddToCart - Customize - QuickAdd - AddTrainingItem - AddEventItem - AddEventSessionItem

AddToCart Quote Fields	Input	String	Y	This argument is used to pass the field values set in the Quote business component. It is a concatenated string of Quote business component field name/value pairs.
				
				The input value is in the following format:
				
				`fieldName1=fieldValue1|fieldName2=fieldValue2|...`
				
				For example, if you set the Description and Discount Reason fields, then the input user property value is as follows:
				
				`Description=Returning customer|Discount Reason=Gold member of the loyalty program`
				
				Note: This only impacts new quotes, and not existing quotes.
				
				AddToCart deconstructs this string and sets the corresponding fields in the Quote business component.

AddToCart Quote Item Fields	Inputs	String	Y	This argument is used to pass the field values set in the Quote Item business component.
				
				The input value is in the following format:
				
				`fieldName1=fieldValue1|fieldName2=fieldValue2|...`
				
				For example, if you want to set the Adjust List Price, Base Price, and Quantity Requested fields, then the input user property value is as follows:
				
				`Base Price=500|Adjusted List Price=450|Quantity Requested=3`
				
				Note: If you set a field that triggers a reprice, such as Quantity Requested, then that field must be set at the end of the string.

AddToCart AutoQuote	Input	String	Y	Used to decide whether to create a new quote automatically if there is no current quote.

ORACLE

Name	Type	Data Type	Optional	Comments
				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. 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. 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.

ORACLE

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 the following table are in the business object, using the Catalog business object as an example.

Business Component	Link	Comment
Internal Product by Price List Optional	None	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	None	Favorite List business component.
Favorite List Item	Favorite List/Favorite List Item	Favorite List Item business component.

ORACLE

Arguments for the AddToFavoriteList Method

The arguments for the AddToFavoriteList method are shown in the following table.

Name	Type	Data Type	Optional	Comments
Action	Input	String	Y	Equals AddToFavoriteList.
Product Id	Input	String	Y	If this input is not empty, add the product specified by this ID to the Favorites list. 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, AddToFavoriteList navigates to this view after completing the AddToFavoriteList action.
ReturnVal	Output	String	Y	Returns Success or Fail. Not used.

Pseudo-Algorithm for AddToFavoriteList

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] = LoginId`;

   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] = input 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.

ORACLE

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 the following table.

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] = LoginId () AND [Quote Type] = LookupValue ('QUOTE_TYPE', 'Persistent 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] = GetProfileAttr("CurrentQuoteId")` to get the current quote. Check the Quote Type field value:

ORACLE

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] = LoginId () AND [Quote Type] = LookupValue ('QUOTE_TYPE', 'Persistent 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 the following table.

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 arguments for the GetCartInfo method are shown in the following table.

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

Name	Value
Quote Template Item|Quote Item FieldMap 100	Product Id|Product Id

ORACLE

Name	Value
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

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

ORACLE

Name	Value
Quote Template Item XA|Quote Item XA FieldMap 107	End

Arguments for the OrderTemplate Method

The arguments for the OrderTemplate method are shown in the following table.

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*
- *SaveCartAsQuote Method*
- *DiscardCart Method*
- *OrderQuote Method*
- *ValidQuote Method*
- *RemoveInvalidItems Method*
- *CreateOrder Method*

ORACLE

- *Reorder Method*
- *Get ShoppingCartStatus Method*
- *MoveToCart Method*
- *RepriceQuote Method*
- *SelectPrimary Method*
- *SetBusCompFields Method*
- *ValidatePayment Method*
- *ValidateAddress Method*
- *SetPendingRequest Method*

> **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*) 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 the following table. For more information, see the Shopping Service business service user properties in Siebel Tools.

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 arguments for the SaveCartAsTemplate method are shown in the following table.

Name	Type	Data Type	Optional	Comments
Quote Name	Input	String	Y	Saves the Shopping Cart as a template with this name.

ORACLE

Name	Type	Data Type	Optional	Comments
Quote Name Field	Input	String	Y	Tells the Shopping Service which field is the quote name. 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. 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.

ORACLE

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 arguments for the SaveCartAsQuote method are shown in the following table.

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. 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. 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.

ORACLE

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 arguments for the DiscardCart method are shown in the following table.

ORACLE

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.
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*), and then RemoveInvalidItems (see *RemoveInvalidItems Method*).

Arguments for the OrderQuote Method

The arguments for the OrderQuote method are shown in the following table.

ORACLE

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	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*), then ValidQuote, and then RemoveInvalidItems (see *RemoveInvalidItems Method*).

- 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 arguments for the ValidQuote method are shown in the following table.

ORACLE

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.
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*), then ValidQuote (see *ValidQuote Method*), 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 arguments for the RemoveInvalidItems method are shown in the following table.

ORACLE

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.
Invalid	Input	String	Y	If equal to N, do nothing. Else loop through each line item to delete invalid items.
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 the following table. For more information, see the Shopping Service business service user properties in Siebel Tools.

ORACLE

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 arguments for the CreateOrder method are shown in the following table.

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	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.

ORACLE

> **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 the following table. For more information, see the Shopping Service business service user properties in Siebel Tools.

Name	Value
Quote|Order FieldMap 100	Contact Id|Contact Id
Quote Item|Order Item FieldMap 100	Class Registration Id| Class Registration Id
Quote Template Item XA|Quote Item XA FieldMap 100	Attribute Id|Attribute Id

Arguments for the Reorder Method

The arguments for the Reorder method are shown in the following table.

ORACLE

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.

ORACLE

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 arguments for the GetShoppingCartStatus method are shown in the following table.

Name	Type	Data Type	Optional	Comments
Quote Type	Output	String	Y	Type of the current quote record. If Persistent Cart, check Status and Count to determine whether the quote is empty or not. 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.
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;

ORACLE

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:

    o eSales - Edit Quote Process

    o eSales - Order Quote Process

    o eSales - Modify Order Process

    o 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 arguments for the MoveToCart method are shown in the following table.

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.

ORACLE

RepriceQuote Method

RepriceQuote reprices the quote.

Preconditions for Calling the RepriceQuote Method

None.

Arguments for the RepriceQuote Method

The arguments for the RepriceQuote method are shown in the following table.

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 arguments for the SelectPrimary method are shown in the following table.

Name	Type	Data Type	Optional	Comments
PickList Field	Input	String	Y	Name of the field that has the picklist.

ORACLE

Name	Type	Data Type	Optional	Comments
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 arguments for the SetBusCompFields method are shown in the following table.

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.

ORACLE

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 arguments for the ValidatePayment method are shown in the following table.

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.

ORACLE

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 arguments for the ValidateAddress method are shown in the following table.

Name	Type	Data Type	Optional	Comments
Street Address	Input	String	N	None
State	Input	String	N	None
City	Input	String	N	None
Country	Input	String	N	None
Postal Code	Input	String	N	None
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

ORACLE

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

Usage:

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