Oracle® iStore and Oracle® iMarketing

Implementation Guide

Release 11*i*

October 2000

Part No. A83719-02

ORACLE®

Oracle iStore and Oracle iMarketing Implementation Guide, Release 11*i*

Part No. A83719-02

Contents

6 Diagnostics and Troubleshooting

7 Profile Options, Accounts, and Forms Settings

Send Us Your Comments

Oracle iStore and Oracle iMarketing Implementation Guide, Release 11*i*

Part No. A83719-02

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most?

If you find any errors or have any other suggestions for improvement, please indicate the document title and part number, and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: Brian.Oden@oracle.com
- FAX: (650) 654-6208 Attn: Oracle iStore and Oracle iMarketing Documentation
- Postal service:
  Oracle Corporation
  Oracle iStore and Oracle iMarketing Documentation
  500 Oracle Parkway, 659603
  Redwood City, CA 94065
  USA

If you would like a reply, please give your name, address, telephone number, and (optionally) electronic mail address.

If you have problems with the software, please contact your local Oracle Support Services.

Preface

Intended Audience

This document is intended for IT professionals who are tasked with implementing Oracle iStore 11*i* and Oracle iMarketing 11*i*.

Related Documents

Check the latest versions of the following documents for additional information on Oracle Applications, Release 11*i*:

- *Release Notes, Oracle Applications Release 11i*
- *Oracle Applications Release 11i Concepts*
- *Installing Oracle Applications Release 11i*
- *Oracle Applications System Administrator's Guide Release 11i*
- *Oracle Applications Implementation Wizard User's Guide*
- *Oracle Applications Product Update Notes, Release 11i*
- *Oracle iStore Release 11i Concepts And Procedures*
- *Oracle iMarketing Release 11i Concepts and Procedures*
- *Oracle General Ledger User's Guide*
- *Oracle Inventory User's Guide*
- *Oracle Order Management User's Guide*
- *Oracle Pricing User's Guide*
- *Oracle Receivables User's Guide*

- *Oracle Master Scheduling/ Planning Guide*
- *Oracle Configurator and SellingPoint Administration Guide, Release 11i and 4.2.2*
- *Oracle CRM:ERP Functional Checklist* (available on Oracle MetaLink)

Typographic Conventions

The following typographic conventions are used in this manual:

Convention	Meaning
italic text	Book titles
`Courier text`	User commands and file content examples
UPPERCASE	Structured Query Language (SQL) commands, initialization parameters, profile options, responsibilities, or environment variables
boldface text	Menu, button, keyboard, and form options
< >	Angle brackets enclose user-supplied names. Note: Do not type the angle brackets.

1

iStore Overview

This chapter provides an overview of the features and architecture of Oracle iStore 11*i*. Topics include:

- Product Overview
- Architectural Overview
- Hardware Requirements
- Software Requirements

Product Overview

Oracle iStore 11*i* provides businesses from all industries with the ability to establish business-to-business and business-to-consumer electronic commerce. The iStore application provides an easy-to-use mechanism for merchants to set up Internet storefronts that capture and process customer orders and to integrate their storefronts with Oracle Enterprise Resource Planning (ERP) applications.

The key features and benefits of Oracle iStore 11*i* include:

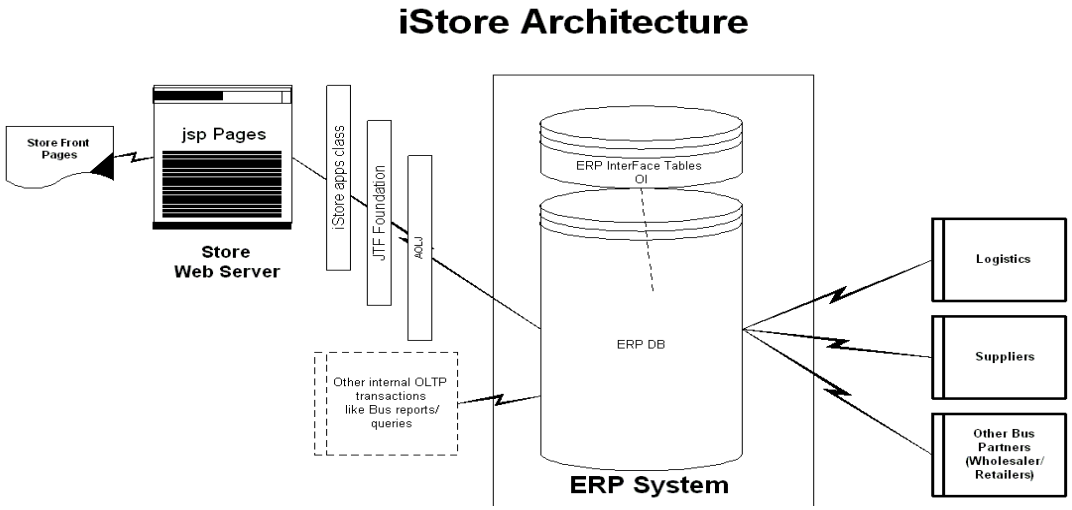
Table 1–1 Oracle iStore Features and Benefits

Feature	Benefit
Rich Product Catalog	Useful and relevant information, easy to find and buy products, and merchandise-driven placement
Configurator	Assisted selling and the ability to sell complex products
Personalization and Recommendations	Cross sell and upsell items and improve visits to purchase ratios
Channel Integration	Leverage assets and processes and achieve a consistent customer experience across contact points
B2B Support	Reach all customer bases as well as manage channel relationships better
Sophisticated Order Capture	Use customer specific pricing, shipping and handling, and tax as well as access to inventory availability
Specialty Stores	Create different stores for different needs, single operations, and reuse of UI data and processes
Self Administration (Business)	Customer manages users and processes as well as lower operational costs for both
Customer Profiles	Personalization and better customer experience
Order Status, Invoices, etc.	Better service and cost efficiency

Architectural Overview

In Oracle iStore 11*i* architecture, data is loaded and retrieved directly from ERP main tables. This functionality provides instant real time data status to web stores and to other integrating systems and applications.

Figure 1–1 Oracle iStore Architecture



Hardware Requirements

The suggested hardware configuration for Oracle iStore 11*i* is a series of web servers in the front and a high performance database server machine in the back end. With global systems, the necessity for high performance database servers is even greater.

Oracle recommends the following server requirements:

- ERP database server machine - high throughput at fast speed (CPU)
- Web servers running Apache for external customers
- One forms server for administration

You can determine the actual sizing of the machines after completing capacity planning.

Specific hardware requirements depend on the particular installation that you perform. The hardware requirements listed in the following table are guidelines only, and assume a single-node Vision demo environment.

Table 1–2 Minimum Hardware Requirements

Hardware	Requirement
CPU	2 CPUs minimum, 4 or more highly recommended
Memory	256 MB minimum, 1GB or more highly recommended
Disk Space	22GB, including 1GB in /tmp (plus an additional 9GB if installing from a staging area)

Software Requirements

The minimum software requirements are listed in the following table.

Table 1–3 Minimum Software Requirements

Software	Requirement
Database	8.1.6 version of Oracle 8i
Middle-Tier	The middle-tier requirements are an Apache 3.0 version web server and Oracle Forms server 6.0 (as a part of ERP 11i implementation). For faster page serving, you can also use a caching server in front of the web server series. The caching server (for example, Calypso) then serves the static content through the cached TCP/IP pockets (or cached pages). An invalidated cache results in service requests being directed to the web servers.

2

Oracle iStore 11*i* Dependency Setups

This chapter describes the setup of applications and other program functions on which Oracle iStore 11*i* depends. Topics include:

Setup Process Flow Diagram

Figure 2–1 Dependency Setup Process Flow (Merchant UI)

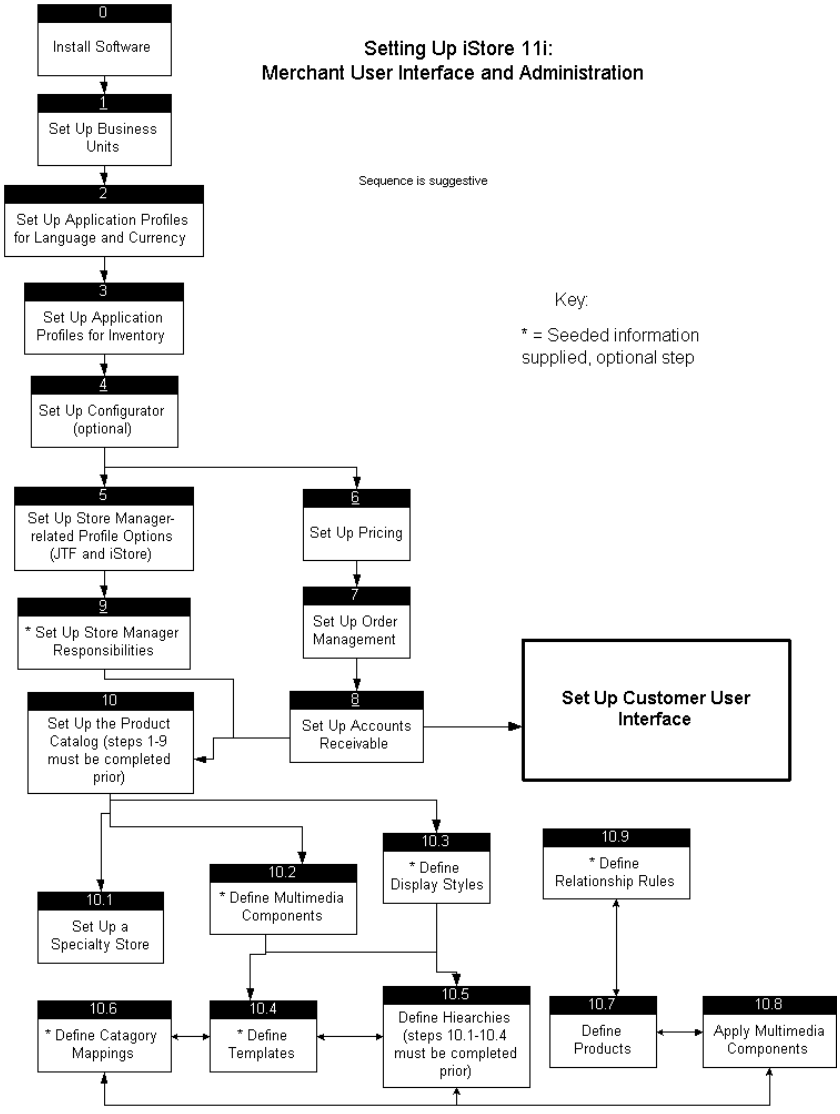
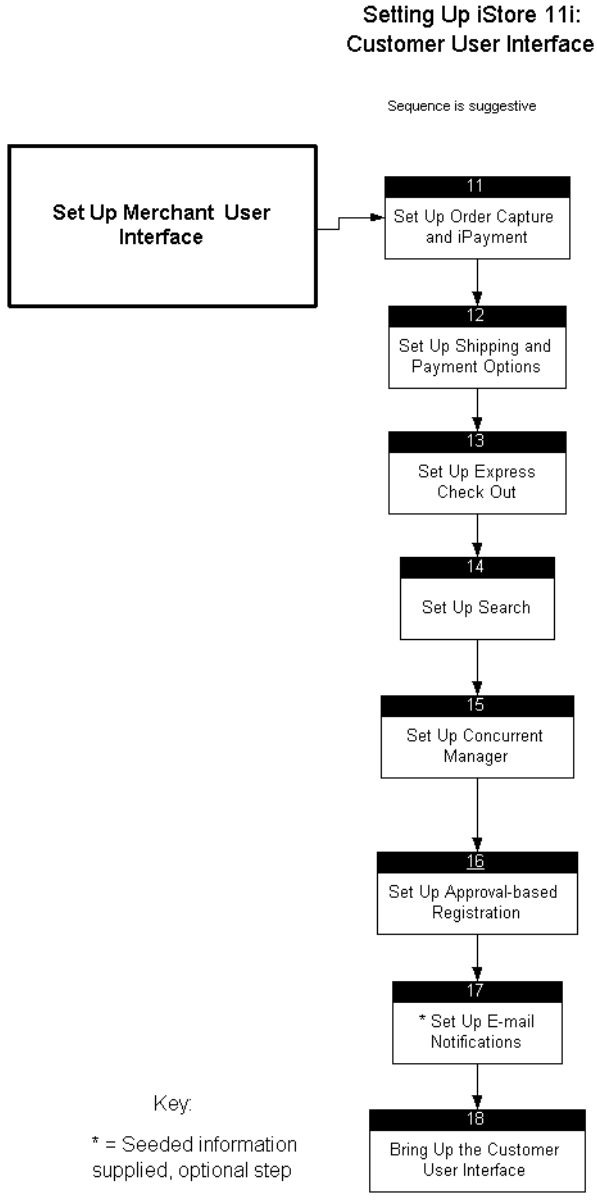


Figure 2–2 Dependency Setup Process Flow (Customer UI)



Dependency Requirements

Oracle iStore integrates with many other Oracle applications modules to provide and extend its functionality. You must setup the mandatory modules before iStore can run. Setting up the optional modules is not required; however, if they are not set up, then the additional functionality provided by these modules will not be available.

Mandatory Modules

Modules listed in this section must be set up in order for Oracle iStore to function properly.

Oracle Application Object Library

Oracle Applications Object Library (AOL) is a required dependency of all Oracle applications modules. Oracle iStore 11*i* uses the AOL to manage responsibilities of store managers as well as customers. You also would define new languages and manage prompts for your store here.

Oracle Inventory

Oracle Inventory is a required dependency for iStore. The following information is set up in Oracle Inventory:

- Category structure to set intelligent defaults for the ways products can be displayed as well as default values for different multimedia components used to display products
- Item information
- Oracle Inventory itself requires at least one inventory organization to be set up and at least one business unit (organization) to be set up. In addition, it requires at least one product catalog group to be set up, even though you may not be using it.

Oracle Order Management

Order Management in turn depends upon Oracle Advanced Pricing and General Ledger. Oracle iStore 11*i* uses Order Management (OM) to keep record of orders the customers placed and pricing of those orders. It does so by using APIs provided by Order Capture (OC). Order Management in turn uses:

- Oracle Accounts Receivable for keeping record of customers, invoices, capturing payments upon shipments

- Oracle Shipping for shipping execution
- Oracle Advanced pricing for determining prices of goods sold

  Note: Order Management in turn requires you to set up the inventory and business units as well as financials-related parameters.
- Oracle Order Capture

Optional Modules

The following Oracle applications modules can be set up to provide additional functionality for your electronic store:

- Oracle iMarketing and Marketing for promotions
- Oracle iPayment to authorization and capture of Credit card transactions
- Oracle Material Requirements Planning to provide product availability information
- Oracle iSupport to provide return authorizations and knowledge base integration
- Oracle Service applications to provide sale of services and products requiring agreements
- Oracle Telephony Manager to process call-me-back requests
- Oracle CRM Business Intelligence to assess the performance of the store
- Oracle Configurator to enable customer configured products and provide guided selling as well as to perform some of the validations of the shopping cart
- Oracle Workflow to send e-mail confirmations to customers
- Oracle Shipping to provide shipment information

Setting Up the Mandatory Dependencies

The following steps indicate the suggested task sequence for setting up iStore's mandatory dependencies:

1. Define and enable language.
2. Define and set up your Business Units, Set of Books.
3. Set up Inventory.
4. Set up Accounts Receivables.
5. Set up Order Management.
6. (Optional) Define more users for store manager.
7. Set up JTF, ASO, and iStore profiles.
8. Create the product catalog.

Using the Oracle Implementation Wizard

Use the Oracle Application Implementation Wizard (AIW) to coordinate dependency setups and identify the steps required to implement the iStore application.

You can use the AIW to see the graphical overview of the steps involved, read online help on set up and open the appropriate forms. You can also document your actions for further reference and review.

Please refer to the *Oracle Application Implementation Wizard User's Guide* for more details.

Setting Up Languages

Oracle Applications 11i enables you to have a multi-lingual set up against one instance. However the set of languages available is dependent upon the character set of the database implementation. The languages enabled determine the set of languages in which the store could be presented to the user.

For a given specialty store, you select languages from those languages enabled at the Oracle Application level. These will be the languages that are going to be available in that specialty store. For details of multilingual capabilities at the store level please see "Creating Specialty Stores", "Defining Multimedia Components", and "Defining Templates" in this document for more information.

Setting Business Units and General Ledger

Because Oracle Order Management and Oracle Inventory require at least one Multi Org and associated set of books, you will need to create at least one business unit. See *Oracle Applications Release 11i Concepts* for more information on business units and multi-org. See the *Oracle General Ledger User's Guide* for steps involved in setting up business units.

Setting Up Oracle Inventory

Oracle Inventory serves as the repository of product items that can be sold through Oracle iStore. You use the Oracle Inventory forms to create new items and then create additional content for the web through the Store Manager. Before you can create items in the inventory system, you have set up and define the structure around it. Refer the *Oracle Inventory User's Guide* for details of inventory set up.

Guidelines

While making decisions about set up, please keep the following guidelines in mind:

- For items that are going to be sold through the web you must set their WEB_STATUS flag to UNPUBLISHED. When you create an item in inventory master you would see this flag grouped under WEB_OPTIONS.
- iStore uses the category structure to keep track of default ways of displaying products as well as default values for multimedia components associated with the product. All items belonging to a category in specified category sets are treated similarly from display perspective. In iStore setup, you set the IBE: Category Set profile option to a category set value that helps iStore differentiate between items based on what categories they belong to within this category set. Items in the same category in this category set are treated as homogenous from display and multimedia default perspective.

  If you are planning to sell items of different types (e.g., books vs. computers) and need to display them differently (i.e., use different templates) then you should create two different categories within this category set. If you do not have the flexibility to do so, then you may specify the display properties at item level. Refer to the "Setting Up the Product Catalog" section of this document for further details.
- Set up flexfields according to your needs. Flexfields are used to capture additional information about items.
- Only customer orderable items can be sold through the store.

- iStore requires one Inventory Org to be identified against which the product catalog is built. Typically this would be the master inventory Org.

Steps

The MTL_SYSTEM_ITEMS table in Oracle Inventory is where product information resides. The following minimum setups are required:

1. Set the following parameter values:
   - Set the value for WEB_STATUS flag parameter.
   - Set the value for the CUSTOMER_ORDER_ENABLE_FLAG parameter.

   These settings are required to display products in the merchant UI.
2. In Inventory, map inventory items to organizational IDs (organizational IDs are set in General Ledger). This mapping determines the inventory items that specific organizational IDs can view.
3. Enter product descriptions.

Setting Up Order Management

iStore uses Order Management to record customer orders and to set up payment options and shipping options and to provide order status and shipping information to customers. Please see the *Oracle Order Management User's Guide* for details of setting up Oracle Order Management.

Setting Up Oracle Pricing

Setting up Oracle Pricing is one of the required steps for Order Management but is being described here separately. For items you plan to sell you must specify the price for that item in at least one price list and make that price list available to customers. For walk-in users, iStore picks up a default price list to show the price of items on the product catalog pages on the store as well as to price the shopping cart. The default price list to be used for different currencies is specified in the Store Manager (see "Creating Specialty Stores").

Using Advanced Pricing, you can also set up promotions and discounts. iStore uses the Pricing engine to determine the best price the customer can get based on the items in the shopping cart and the customer. You set your pricing rules in Advanced Pricing Module. However, Oracle iStore does not support pricing attributes, nor does it support customer-asked qualifiers (promocodes) in this release.

Steps

The following minimum setups are required:

1. Set Qualifiers in QP. Qualifiers are used to calculate item prices.
2. Set Modifiers in QP: Modifiers determine discounts or can be used to calculate shipping costs.

Refer to the *Oracle Pricing User's Guide* for more information on Oracle Pricing.

Pricing Setup Example (Define Simple Freight Charges)

This example shows how to set up simple fixed-amount freight charges in iStore:

Steps

1. Go to Oracle Forms, log in as SYSADMIN / SYSADMIN, choose the Oracle Pricing Manager responsibility.
2. Go to **Modifiers > Define Modifier**. (If you don't have the Oracle Pricing Manager responsibility, grant it to the SYSADMIN user first.)
3. Click **LOV** for Type.
4. Click **Freight and Special Charge List**.
5. Enter some identifier for the modifier in the Number field (e.g., IBEFR01).
6. Enter some description in the Name field (e.g., IBE Freight Modifier).
7. Click in **Modifiers Summary > Modifier No**, and enter a number (e.g., 1).
8. Click in **Level**, select modifier level (e.g., Line).
9. Click in **Modifier Type**, select **Freight/Special Charge**.
10. Enter a Start Date and an optional End Date.
11. Scroll right. Click in **Pricing Phases**, choose **Line Charges** from the LOV.
12. Click on the Discounts / Charges tab.
13. Click in **Charge Name**, select **Freight Costs** from the LOV.
14. Click **Application Method**, select **Amount**. Enter a per-line freight cost dollar amount (e.g., 3.00).
15. Save the form.

Setting Up Accounts Receivable

iStore uses the Accounts Receivables (AR) module to record customer information as well as tell customers about their invoices and payments made. For details of setting up AR please see the *Oracle Receivables User's Guide*.

The TCA / AR schema is where customer registration information is maintained. At a minimum, you need to set up the following:

- Address Validation
- Tax Codes and the Default Tax Code

Setting Up Order Capture

Oracle iStore 11*i* uses Order Capture to integrate with Order Management (OM) and Accounts Receivable (AR). Once you have set up AR and OM you need to only set up certain profile values. Those profile values are described in Chapter 7, "Profile Options, Accounts, and Forms Settings".

Setting Oracle Store Manager Users

Oracle iStore 11*i* ships with a default seeded store manager user (`ibe_admin` with password = `manager`. You are strongly encouraged to change the password upon installation. This user is granted a default responsibility (IBE_ADMINISTRATOR) through which the user gets all the required menu and privileges to manage the store. The menu assigned to this responsibility is called IBE_ADMIN_MENU.

If you need to create more users you can create them by using the FND_USER forms in the AOL module. If you need to change / modify the functions the user can do you can create a new responsibility for the user. For the new responsibility you can assign the default Menu (IBE_ADMIN_MENU) and still remove access to some of the tabs ("functions" in AOL terminology) or you can create a new menu using the Oracle iStore 11*i* functions. All Oracle iStore 11*i* functions can be found by searching for `IBE_`

See the *Oracle Applications System Administrator's Guide Release 11i* for more details on AOL and managing responsibilities.

Optional Dependencies

In addition to mandatory dependencies, Oracle iStore 11*i* also depends upon the following modules to provide additional functionality:

- Oracle Planning
- Oracle Configurator
- Oracle iMarketing
- Oracle iPayment
- Oracle Telephony Manager
- Oracle Workflow
- Oracle Shipping

Setting Up Oracle Planning for ATP

If you want to provide inventory availability information to your customers, then you must set this up. As part of the rules for determining availability you can provide sourcing rules that encompass orders already placed as well as open purchase orders, etc. For more details, refer to the *Oracle Order Management User's Guide*, the *Oracle Master Scheduling/ Planning Guide*, and *Oracle Inventory User's Guide*.

Setting Up Oracle Configurator

Oracle Configurator is used to create product models and to help the buyer assemble related and dependent products in the shopping cart.

The configurator developer UI helps creating the product models for the dependent and related products, to build rules around the products.

The product models are imported initially from BOM models. The developer UI can be used to create a tree structure for the product model. For example, if a customer wants to build their own laptop from a store of electronic items, then the developer UI of the configurator can be used to help define the structure and the dependent (mandatory and optional) products (e.g., a 15" or 17" screen with 16MB or 32MB RAM and 6GB or 8GB drive, etc.).

When the customer is browsing items on the featured section, if the item has the "MODEL" flag set in Oracle ERP applications, a "Configure" button appears on the store UI. For every item, Oracle iStore 11*i* sends an init message call to configurator

to find whether a configurator UI is associated with the item. If a Configurator UI is associated with an item, the store UI adds the Configure button or a link.

When the user clicks the Configure button, the store sends another init message to populate the configurator UI with item id. The configurator UI appears in the frame and the related / dependent products can be assembled for placing the order. There is a top menu bar with buttons such as Save / Done. Once the customer has finished building the list of selected items for the order, clicking the Save button places all the items in the shopping cart.

For more information, see the *Oracle Configurator and SellingPoint Administration Guide, Release 11i and 4.2.2* for detailed setup documentation.

Changing the Configurator UI The Configurator UI appears in the frame provided on the iStore featured section. This configurator window is created with DHTML or Java Applet. The look and feel are similar to the Oracle iStore 11*i* UI. If you want change the Configurator window, you can modify the HTML templates for Configurator. These templates are loaded in Oracle iStore 11*i*'s HTML directory, on the same server, by default. The Configurator data is stored in the CZ tables in the APPS schema.

Profile Setup for the Configurator UI The only setting required for Oracle iStore 11*i* to get the Configurator UI is a URL that handles all interaction between the client and the server. For example, the URL could be

`http://apps-server-host/apps/cz/oracle.apps.cz.servlet.UiServlet`.

The server and directory structure are installation information that the calling application must read, but the `oracle.apps.cz.servlet.UiServlet` portion is always the same. This URL is read from the CZ_UIMGR_URL profile variable.

The setup needed to run the Configurator UI from the Oracle iStore 11*i* customer UI is the setting of the above mentioned profile variable.

Setup Steps

Once the rapid install has been done, the Configurator servlet is setup and tested, and the UI built, complete the following steps:

1. Using the Forms application, every item that can be shown for this configured item (from the root model item, every option class, and every leaf node) must be:

- Web Status = “Published” and Orderable on the Web = “Y” (checked) this is done at: **Inventory responsibility >Items>Master Items** (choose the org) **> Web Option** tab.
- Added to the price list that will be used by the store/customer (even if it's there with a zero price)
- For the IBE_CUSTOMER responsibility, the profile option BOM: Configurator URL of UI Manager must be set to be:

  `http://<machine>:<port>/servlets/oracle.apps.cz.servlet.UiServlet`

2. Using the merchant UI, the model item has to be added to some part of the catalog in order to be displayed.
3. The .dbc file to connect to the database must be in place in the `secure` directory under the directory defined as `fndtop`.
4. The jserv.properties file must have the template url defined as follows (these urls must be able to be resolved when entered into a browser):

   ```
   wrapper.bin.parameters=-Dcz.uiservlet.templateurl=http://<machine>:<port>
   /OA_HTML/US/czFraNS.htm

   wrapper.bin.parameters=-Dcz.uiservlet.templateurl.ie=http://<machine>:<port>
   /OA_HTML/US/czFraIE.htm
   ```

Testing the Configurator Setup

1. Enter the following url in the browser:

   `http://<machine>:<port>/servlets/oracle.apps.cz.servlet.UiServlet?test=version`

   The browser should return the following statement:

   `Using configuration software build: 11.5.1.14.27 Expecting schema: 14c`

   This will inform the merchant whether the configurator middle tier servlet is up and running or not.
2. Edit the fields in the Configurator Standalone Test page, open it in a browser, and click on the **Launch DHTML** button. The UI should appear.

Setting Up Oracle iMarketing

You can use Oracle iMarketing to personalize the store and make recommendations. You create postings in iMarketing, create rules that determine the content for a given posting, and then modify Oracle iStore 11*i* templates to make reference to posting tags.

See Chapter 5, "Implementing Oracle iMarketing" in this document for more details.

Setting Up Oracle iPayment

If you are planning to provide credit card payment options, then you must set up Oracle iPayment to perform credit card authorization and fund capture. Authorization itself can be set up to be done when the order is being placed or can be set to be deferred to do it at a later time. Refer to Chapter 4, "Oracle iStore 11i Setup and Administration" for instructions on configuring the behavior.

Also see the *Oracle iPayment Implementation Guide*, the *Oracle Order Management User's Guide* and the *Oracle Account Receivables User's Guide* for additional set ups required for iPayment. In addition to iPayment setup, you will need to set up the Oracle iPayment system itself to talk to provider networks.

Setting Up Oracle Telephony Manager

Oracle iStore 11*i* provides call-me-back functionality whereby customers can access a call being placed in the call center. Please see *Oracle iSupport Concepts and Procedures* for details.

Setting Up Oracle Workflow

Oracle Workflow is used to send e-mail confirmations to customers and e-mail alerts to parties that share a shopping cart. At a minimum, you must set up message text in Workflow for Oracle iStore 11*i* e-mail notifications. See the *Oracle Workflow Guide* for more information.

Setting Up Oracle Shipping

The Oracle Shipping module must be in place to enable post-order tracking and shipping detail views in Oracle iStore 11*i*. At a minimum, you must specify shipping methods and descriptions in Oracle Shipping. See the *Oracle Shipping Execution User's Guide, Release 11*i for more information.

3

Installation and Dependency Verification

Before you start configuring the profile options and creating the product catalog in Oracle iStore 11*i*, check that you have correctly completed the following prerequisites.

Checkpoints

1. Verify that the installation and middle tier setup have been done correctly.

   Once the Rapid Installer Wizard finishes the installation, verify that the proper installation and configuration of the following components:

   - **Apache Server**: Go to http://<host>:<apache port>/apachedocs/. You should see the Apache Server documentation page.
   - **Apache JServ:** Go to http://<host>:<apache port>/servlets/IsItWorking. You should see a message reassuring you that Apache JServ is working.

2. Verify that the ERP applications are installed and functioning properly. Refer to the *Oracle CRM:ERP Functional Checklist* document available on Oracle MetaLink for a description of the tasks required.

3. Verify that the setup dependency has been done correctly.

   Place an order through Sales Order bench. If it goes through correctly then your core dependency has been done correctly. To find how to place an order, refer to *Oracle Order Management User's Guide*.

4

Oracle iStore 11*i* Setup and Administration

This chapter describes the setup and administration of Oracle iStore 11*i* after you have verified your installation and dependency setup. Topics include:

- Overview of Creating Your Store
- Setting Up the Product Catalog
- Creating Specialty Stores
- Defining Hierarchies
- Setting Up Product Searches
- Setting Up Credit Card Payments in iStore
- Testing the Store
- Issue Reporting Guidelines

Note: When using the Merchant UI, ensure that cookies are enabled.

Overview of Creating Your Store

Use the following procedure as an example of the sequence of steps you can use to create your store.

Prerequisites

Identify the ways in which you will display products in the store. Plan your page designs and divide the designs into common components that you can make into templates. Invent a name for each possible template to facilitate planning and communication of designs. Oracle iStore 11*i* ships with templates that allow you to develop the initial store. You can customize templates now, if you need to, as part of the design or as a revision to the templates later in the setup cycle.

Steps

1. Set up at least one specialty store by choosing **Specialty Stores** in the Setup tab.
2. Define the types of media you will use when building your store by choosing **Multi Media Components** in the Setup tab. Examples of media types include small gif, large gif, short description, and 10 second audio file.
3. Enter default source files for your multimedia components using **Multi Media Components** in the Setup tab.
4. Define each type of display template you identified above by choosing **Display Styles** in the Setup tab.
5. Enter default source files for your display styles using Display Styles in the Setup tab.
6. Create templates for pages and for blocks within pages using your chosen web authoring application.
7. Record and organize your template files using the Template tab. (Do not yet assign categories to the template.)
8. Create one overall hierarchy for your products, including grouping them into sections, using the Hierarchy tab.
9. Choose the section from your hierarchy to be the root node for your specialty store by choosing **Specialty Stores > Store Flags** in the Setup tab.
10. Create media files outside of Oracle iStore. Media include graphics files (.gif), text files, audio files, video files, and so on.
11. Assign templates and multimedia to product categories in the Category tab.

12. Build your product catalog in the Product tab and assign templates.
13. Record and organize your media files using the Multimedia tab.
14. Build product relationships for merchandising using the Relationship tab.

Setting Up the Product Catalog

Setting up the product catalog involves the following considerations.

- Designing screen flow and navigation, also called developing hierarchy structures, the principles and philosophy behind the navigation paths.
- Determining items to be sold, their display features, and configuration options, also called detailing fields for inventory and configurator.
- Determining types of data required, for example, books = title, author, publisher; jacket = size, color, men's, women's. Use the flexfield structures in Inventory to store and sort data accordingly.
- Designing the display appearance for different product types. This process determines the number and type of product templates required. For example, perhaps all music products list the artist first and then provide a link to an audio clip, but all clothing products list the clothing type, for example jacket, first followed by a static graphic of the item. Oracle iStore 11*i* ships with the assumption that all product types appear the same on the Customer User Interface (UI).
- Creating template text for product types. Text imbedded in a template makes that template specific to the given product type. For example, the word "Artist" in front of the flexfield where a performer's name is to appear can only be used for the compact disc product category. Imbedded text must be manually translated and saved in the required multiple languages as additional template types. Oracle iStore 11*i* does not translate template text. Alternatively, templates using generic terminology can be more easily applied across product types. For example, using the term "Lead Performer(s)" as a flexfield label could apply to both compact disc and videotape product categories. Providing no flexfield labels in a template allows templates to be most broadly applied across product types.

Prerequisites

- Products must be loaded into inventory before they can be imported into Oracle iStore 11*i*. The publish / unpublish flag identifies the store-bound products.

- Profile options must have been set.
- Shipping options must have been seeded into Oracle Shipping.
- Payment options and setup must have been seeded into iPayment.
- Store layout must have been determined with the following considerations:
  - Site appearance must have been decided.
  - Hierarchy of products, sections and specialty stores must have been identified. If the site has to be deployed in multiple languages and currencies, you may create a specialty store for each language.
  - Templates associated with each product, section and specialty store must have been identified.
  - Templates and sub-templates must have been identified.
  - At the implementation level, the mapping between templates and source files must have been decided.
  - Source files (physical templates) must have been created by the UI implementation team with stubs for the dynamic elements, along with the multimedia components to be displayed on the site.
  - Templates must have been populated with the dynamic JSP elements calling Oracle iStore 11*i*, using the templates shipped with Oracle iStore 11*i* as a model.

Steps

Following is the suggested sequence of steps for setting up Oracle iStore 11*i*. Alternatively, you can use the Merchant UI navigational sequence of creating the Specialty Store first.

1. Define JTF profiles.
2. Define IBE Profiles.
3. Run concurrent jobs.
4. Set up shipping options:
5. Set up payment options.
6. Set up product search.
7. Create multimedia.
8. Create / modify custom templates.

9. Identify templates.
10. Define display styles.
11. Create content.
12. Define multimedia components.
13. Create store sections (hierarchy).
14. Create store hierarchy.
15. Create specialty store.
16. Create product relationships.

Creating Specialty Stores

A specialty store is any web store. You can create multiple stores, for example a main store, a store for one large customer, a holiday specials store, and a store that requires registered users. You must create at least one store. Use the following procedure to create a store.

Prerequisites

Check that you have set up the following prerequisites:

- Business units in Oracle General Ledger
- Price lists and currencies in Oracle Order Management
- Language in AOL

> **Note:** If the information for these prerequisites is unknown, you can continue with the setup now and revise this information later.

After you build your product hierarchy you can return to this tab and specify the root section from your hierarchy for this specialty store in the Store Flags page.

A default language is set for the store. However, a customer can then choose a preferred language from those offered. Once a registered customer selects a preferred language, the store defaults to that preferred language each time the customer enters.

You can choose to make the store active at a later time.

You must select a price list for each type of customer on the Currencies and Price Lists page. Only maximum order limit is optional.

To create a new speciality store, click Create or click the name of the specialty store that you want to modify. You can reach this section from the **Merchant UI > Setup > Specialty Stores** section.

> **Note:** Each Specialty Store must have a root node, which has not been created. As a work around, leave the root ID blank and proceed to the step "Choose root node," and then return.

Steps

1. In the Setup tab, choose **Specialty Stores**.

   The Specialty Store page lists existing store names and descriptions.

2. Click **Create**.

   The wizard for creating a specialty store starts.Enter the basic information for the store, including name and default language, and click **Continue**.

   The Supported Business Units page displays available business units from Oracle General Ledger.

3. Select a default business unit, optionally select other business units, and click **Continue**.

   The Specialty Store Flags page appears.

   This section is used to define the profile for whether the specialty store will have:

   - Walk-in customers allowed
   - ATP Enabled for inventory
   - Root Section - Name of the root store (required).

4. Choose whether or not this store is active, provides available to promise inventory information to the customer, and allows walk-in customers who do not log in, and then click **Continue**.

   The Currencies and Price Lists page displays available currencies.

5. Select currencies and price lists for each selected currency.

Assign the Price lists for Walk-in customer, Registered customer and Business customer. This screen will also be used to set the currency for the specialty store and the item orderable limit. Multiple currencies and languages can be selected for a specialty store. User's preference will define which currency and language is to be used for the store. When the user preference is not set, default language and currency settings will take effect. The default language and currency settings should be done prior to setting the store flags.

6. Enter a maximum order limit for each selected currency.
7. Click **Enable As Store Site**.

   The new specialty store is saved.

Cataloging Multimedia

Multimedia consist of files used to present content on a web page to your customer, such as graphics, text, audio, and video. Use this procedure to add multimedia names and to catalog available media files mapped to the multimedia name.

Naming Multimedia

The multimedia name is the catalog name that is easy to communicate and use when planning your page designs. An example is *CompanyLogo*.

Every multimedia name is given a programmatic access name that is short, unique, and not as descriptive. The programmatic access name is used to display that multimedia file in your web page, if you want to refer to it directly. It is not translated. An example is *clogo*.

The multimedia name and programmatic access name represent several source files. You assign each source file to combinations of specialty stores and languages. The following table lists examples of file names for the example name *CompanyLogo*.

Table 4–1 Sample Media File Names

File	Specialty Store	Language
clog1f.gif	specialty store 1	French
clog1e.gif	specialty store 1	English
clog2f.gif	specialty store 2	French
clog2e.gif	specialty store 2	English

The result is that if a French customer enters your specialty store 1, the store displays the logo file clog1*f*.gif. If an English customer entering the same specialty store 1 sees clog1*e*.gif.

You can search for multimedia more easily if you enter keywords for the multimedia.

Prerequisites

- The default language must have already been defined.
- At least one speciality store must have already been created.

Steps

1. In the Multimedia tab, conduct a search for media that is already cataloged and available to use in your store.

   The Multimedia page lists the multimedia that match your search criteria along with the access names assigned to the multimedia, keywords, descriptions, and the default source files to use for all specialty stores and languages.

2. Click **Create**.

   The Multimedia Details page appears.

3. Enter the media detail information.

4. Define the programmatic access name, which is the name by which the media will be accessed from the template.

5. Define a common name to which the media can be referred.

6. Optionally, define a media description.

7. Define the default source file which contains the media content. Define the location of the file relative to the OA_Media set up for example, `/OA_MEDIA/product.gif`. This default source file will be used by Oracle iStore 11*i*, unless a specific mapping for a speciality store or non-default store language is preferred by the customer. If only one language or specialty store is defined or if no specialty store has been created, use the defaults to continue with Oracle iStore 11*i* set up.

8. Optionally, click **Add** to provide files for the same multimedia object in different languages and specialty stores.

   The Source File Details page appears.

9. Enter the name of a media source [physical] file, such as a graphic file, that you want to display on a web page for the media name you are creating, for example, `/OA_MEDIA/video.jpg`.

10. Add each specialty store and languages where you want the new source file to appear, then click **Update**.

    The relationship between the new media name, source files, specialty stores, and languages is saved.

11. Repeat step 4 for each source file you want to add.

12. In the Multimedia tab, choose **View All Mappings**.

    The View All Mappings page displays each source file name and its relationship to specialty stores and languages. This step is highly recommended, but not required to continue the store set up.

Defining Multimedia Components

Multimedia components define the types of media objects available to display on a web page, such as a certain size picture, short text description, or audio of a certain length. Use this procedure to define multimedia components that you want to assign to sections, categories, or products.

Prerequisites

You can select a default multimedia only after you have cataloged multimedia. If the default information is unavailable, you may proceed with the setup process and select a default multimedia at a later time. However, if a multimedia association is requested for any product or section with that multimedia component and an association is not specified between the product or section and the multimedia component, Oracle iStore 11*i* will use the default multimedia defined at the store level (profile value). To avoid the error, you can also use the multimedia component's seeded values.

Seeded Values

- STORE_PRODUCT_LARGE_IMAGE
- STORE_PRODUCT_SMALL_IMAGE
- STORE_SECTION_SMALL_IMAGE

Steps

1. In the Setup tab, choose **Multimedia Components**.

   The Multimedia Components page displays a list of existing media components and default source files that the store uses if no file is assigned to the product or category.

2. Click **Create**.

   The Multimedia Component Details page appears.

3. Assign names and descriptions to your multimedia component.

4. Optionally, click **Go** to select a default multimedia for this component.

5. Click **Update**.

   The multimedia component information is saved. The multimedia components appear in the Product and Category tabs where you choose a multimedia name to correspond with the component for the product or category. If no multimedia name is selected for both product and category, then the default multimedia that you entered in step 4 is used on any store web page for that product or category when it is associated with this component.

Defining Display Styles

When you design your pages, you can specify in a display style how to present products. For example, you can use one display style to display product A on a special sale page containing multiple products, and a different display style to display product A on a page detailing product information.

Prerequisites

You can select a default template only after you have cataloged templates. If the information is unavailable, you may proceed with the set up process and select a default template at a later time. However, if a template association is requested for any product or section with that display style and an association is not specified between the product or section and the display style, Oracle iStore 11*i* will use the default template defined at the store level (profile value).

To avoid the error, you can also use the seeded values for display styles:

1. Go to Setup Display Styles.

2. Review the seeded values. Optionally, create more.

Seeded Values

- STORE_FEATURED_PRODUCT
- STORE_PRODUCT_DETAIL
- STORE_PRODUCT_SMALL_DESCR

Steps

1. In the Setup tab, choose **Display Styles**.

   The Display Styles page displays a list of existing display styles.

2. Click **Create**.

   The Display Style Details page appears.

3. Assign names and descriptions to the display style.

4. Optionally, click **Go** to select a default template for this display style.

5. Click **Update**.

   The display style information is saved. The display styles appear in the Product and Category tabs where you choose a template name to correspond with the display style for the product or category. If no template name is selected for both product and category, then the default source file you entered in step 4 is used on any store web page for that product or category.

Defining Templates

Define templates to control the way the store looks through the use of Java Server Pages (JSP) which combine Application Programming Interfaces (APIs) to call the dynamic data and HTML to present the static data.

Verifying Templates

To verify templates, click on Templates, and search for STORE_HOME. It should be set to ibezhome.jsp.

Display Manager/Template Manager: Once customized templates are created, the templates are to be mapped in Merchant UI. The display manager screens in Merchant UI are used to give a logical name to a template.

Note: Leave this blank if you are running Oracle iStore 11*i* out of the box.

The first screen in the “Templates” tab, lets the user enter the basic details of the template. Once these details are entered, the template then can be assigned to a category / section / item and can be mapped based on the display style and language settings, on subsequent screens.

Template Assignments

- **Item** - For a particular item to be displayed differently based upon a display style, like a "Featured Product."
- **Category** - For a specific Display Style
- **Section -** For example, Database section can have a different template than CRM section.

When an item is displayed, Oracle iStore 11*i* process searches for the template assignment on item level. If there is no item-specific template defined at this level, it searches for the Category level template. If there is no template at this level, it searches for the Section level template. If the section level template is not defined, the store level template is used.

After the process defined above has retrieved the logical name for the template, the source filename of the template is determined by the Logical Name + Specialty Store +Language mapping.

Table 4–2 Template Names, Keywords, and Descriptions

Template Name	Programmatic Access Name	Keywords	Description	Displays	Default Source File
ADDRESS SELECT	ST_ADDR_SL	Address	Select a particular address	OTHERS	ibeaads1.jsp
AGREEMENT	ST_AGGR	Agreement	Agreement for consulting	FEATURE	ibeagreem.jsp

To set the mapping for the template language and specialty store combination choose **View All Mappings > Specialty Store and Language Mappings >Template Name**.

Table 4–3 Template Language Mappings

Specialty Store	Languages
Oracle Store External	ALL

To assign the categories to the template, choose **Template > Categories > Template Name**.

Lookup: Name

Table 4–4 Template Categories

Category Name	Description
MISC.MISC	Miscellaneous

When you assign the categories to the template, you can view the template details for a category from the Category tab by choosing **Category >Category Name >Templates Assigned**.

Cataloging Templates

Templates are put together to create a web page. When you design your pages, you use designs that consist of common components that you make into templates. Use this procedure to create names and programmatic access names for templates that you design and assign to specialty stores and languages.

This step involves identifying the JSP templates that are needed to run Oracle iStore 11*i*. The out-of-the-box application comes packaged with a complete set of templates needed to run the store. If you want to expand the functionality or wish to customize the pre-packaged templates, then you need to identify the flow of the application and identify the JSP templates needed to implement the flow.

There are two steps required for setting the templates for the store. The first step is to create or modify the templates. The second step is to set up the mapping through the display / template manager functionality that is provided in the Merchant UI.

This section describes the set up of the templates, which are available as standard installation or are created new. This step involves adding new templates to replace or add to the templates that are already available in Oracle iStore 11*i*. It involves adding the name and description for these new templates and also specifying the different source (physical) JSP templates that will be used at run-time based on language and the specialty store.

The template name is the catalog name that is easy to communicate and use when planning your page designs. An example is `ProductHome`.

Every template name is given a programmatic access name that is short, unique, and not as descriptive. The programmatic access name is inserted into your web page or template. It is not translated. An example is `phome`.

The template name and programmatic access name represent several physical files. You assign each physical file to combinations of specialty stores and languages. The following table shows an example of files for `ProductHome`.

Table 4–5 Example of File Assignments

File	Specialty Store	Language
hom1f.jsp	specialty store 1	French
hom1e.jsp	specialty store 1	English
hom2f.jsp	specialty store 2	French
hom2e.jsp	specialty store 2	English

The result is that if a French customer enters your specialty store 1, the store displays the home page file hom1f.jsp. An English customer in the same specialty store 1 sees hom1e.jsp instead.

You can find templates more easily in a search if you enter keywords for the template.

Creating and Modifying Templates

You can create new templates to replace or add to the templates already available in the base product. Creating templates involves adding the name and description of these new templates and also specifying the different physical JSP templates to be used at run-time based on language and the specialty store.

Major skills required to modify the templates are HTML and JSP (Java Server Pages). The templates are in the HTML directory base. All ibem*.jsp templates are for Merchant UI, all ibez*.jsp templates are the included templates for customer UI. Users can create new templates in this directory.

The Oracle iStore 11*i* seeded templates are developed in JSP, which is basically a dynamic HTML web page. JSP is Sun's answer to Microsoft's ASP (Active Server Pages) where the script language can be embedded in the HTML content for making the dynamic web pages. ASPs embed Visual Basic language scripts in HTML with the special tags and JSP embeds Java language methods in the HTML

content to generate dynamic content on the web page. The structure of a JSP page is demonstrated in the following HTML example.

```
<HTML>

<% import="oracle.apps.ibe.util.*" %>

....

....
```

`<P> Name : <% = customer.getName(12334) %>` Where `customer` is a Java class on the server and `getName` is a public method in the class to retrieve the customer Name.

`<P> Picture: <IMGSRC = "<%= customer.getPict(12334) %>">` This step can retrieve the image file name from the customer Java class on the server.

After making the changes to the templates, you can pre-compile the template to check for compilation errors and to increase the speed of the initial loading.

API Documentation

To make changes to the UI or page displays, you must have complete knowledge of the APIs being called from the JSP page. These APIs are the key for displaying data on the store pages. These are the application objects / beans. Customers and users cannot modify these class files.

Check Oracle MetaLink for the latest Oracle iStore Javadoc.

Modifying the UI

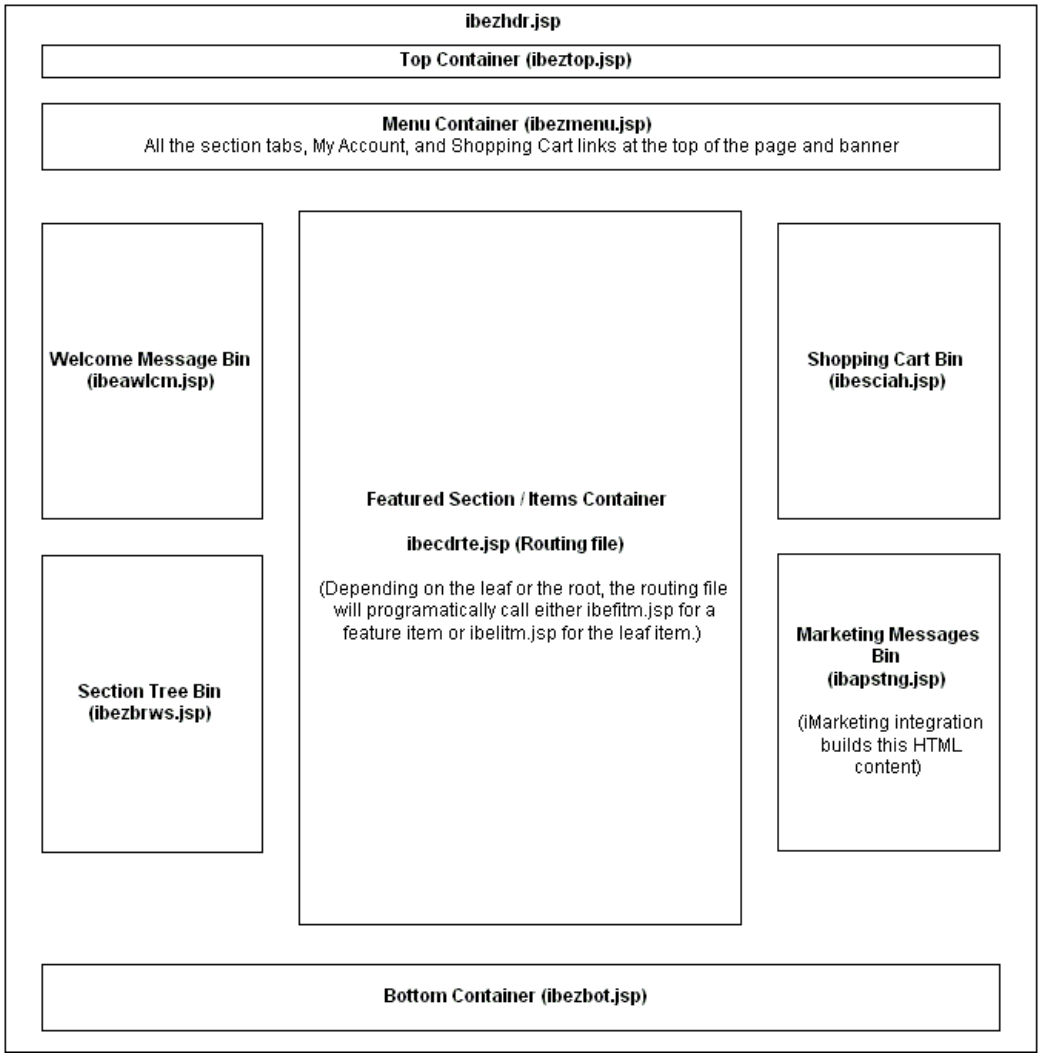
The Oracle iStore 11*i* page is sectioned into various information containers, also referred to as *bins* or place holders. These bins hold the content specific information and display it logically on the page. You can modify the text in the bin and the layout of the bins to change the look and feel of the UI.

Changing the Layout of Bins You can either change the bin placement or remove a bin from the UI by modifying the home page ibezhome.jsp and other corresponding jsp files. Take special care when making these changes, which affect the HTML.

Changing the Text in Bins The text of the bins (for example, Welcome Message Bin, Shopping Cart Bin, Section Tree Bin) comes from the Messages stored in ERP. To change the text, log in to ERP Forms as Application Developer responsibility, choose the Messages menu option, and search for IBE% messages. You can modify

these messages to change the text in the bin. You can find a specific message name by viewing the respective bin JSP file.

The following diagram shows a layout of the bins on the store page. The layout is the default layout that comes with a standard installation.



Seeded Values

See the "Creating and Modifying Templates" section.

Prerequisites

- At least one specialty store must have already been created.
- At least one language must have already been defined.

Steps

1. In the Templates tab, conduct a search for templates that are already cataloged and available to use in your store.

   The Templates page lists the templates that match your search criteria along with the programmatic access names assigned to the templates, keywords, descriptions, and the default source files to use for all specialty stores and languages.

2. Click **Create**.

   The Template Details page appears.

3. Enter the template detail information.

   a. Name: The name by which the template is referred to from the design, the common name.

   b. Programmatic Access Name: The name by which the template is referred to in the JSP.

   c. Description: Optionally, enter a description and key words.

   d. Default source file: The JSP to be used as the default if no non-default language or specialty store mapping is defined.

   e. Define whether the template is to be used to display a section category or other items. If the template will display a product, choose Category from the drop down list.

4. Optionally, click **Add** to provide files for the same template in different languages and specialty stores. See the Guidelines below for details.

   The Source File Details page appears.

5. Enter the name of a JSP source (physical) file that you want to use for the template name you are creating.

6. Add each store specialty store and language where you want the new physical file to be used for this template, then click **Update**.

   The relationship between the new template name, source files, specialty stores, and languages is saved.

7. Repeat from step 4 for each physical file that you want to add.

8. In the Templates tab, choose **View All Mappings**.

   The View All Mappings page displays each physical file name and its relationship to specialty stores and languages. This step is highly recommended, but not required to continue the store set up.

9. Optionally, if the template you created is meant to display product categories, you may optionally assign it to categories at this point by going to the Category tab to assign it. If you choose to assign the template to a category here, click on the **Categories Assigned** and select the categories to which the template is applicable.

Defining Hierarchies

To define hierarchies, create one overall hierarchy that contains products from Oracle Inventory that you group into sections. Associate your specialty stores to a portion of the overall hierarchy or to the whole hierarchy itself by setting up their root section to point to a section. The hierarchy determines the browsing experience of the customer and what products are featured at different levels in the store. When users come to a specialty store, they see the hierarchy starting from the root node of the store. You can choose not to show a particular section in a specialty store even though the given specialty store might be pointing to an ancestor of the store. Your template design determines how to manifest the hierarchy for the user. In the out-of-box store templates, the top level appears as tabs while the lower level appears as browse bins on the store pages.

Set up the top level sections first. For each top level section, create as many subsections or children as you wish. The levels of sections are driven by the design. You can create and revise new templates at any time.

You can assign products that belong to the hierarchy in this procedure or when you work on products. Similarly, you can create groups of featured products at any level in the hierarchy by creating a subsection of type featured in that section. Then in the template for the section you can explicitly show the products in that section. The products in a section are shown by using the display style that you specify for that section.

Out of the box, if a section contains other subsections, show the featured products in the middle and show the subsection on the left browse bin. If the section contains only products, show the list of the products in the middle.

Prerequisites

- Profile set up for IBE:Item Validation Organization (see definition for IBE_ ITEM_VALIDATION_ORGANIZATION profile).
- Optionally, you may also have to set up IBE: Default Section Template. See step 2.

Steps

1. In the hierarchy, select the node below the node that you want to create, and choose **Create**. (If you want to modify existing information, click on the node.)
2. You will see Basic Information Screen. Fill in the details as follows and choose **Continue**.
   a. Name: The name of the section.
   b. Section Code: If you want to refer to the section in templates explicitly by name, use the section code for the section.
   c. Basic And Long Description: If you want to show content about the section in the store, fill in this.
   d. Keywords: For search purpose (not currently used).
   e. Display style for displaying the products in the section.
   f. Template for displaying this section. Several sections can share the template. If you do not provide a value here then value for IBE:Default Section Template is used.
   g. Start Date Active and End Date Active: To specify the tiem when this section will be active.
3. You will now provide information about multimedia components specific for this section. This will be used if you want to show / associate content with sections too. For a given component, hit go to search the multimedia catalog and if not found then go and create one and then associate with the section.
4. Hitting continue after the next steps will lead to a screen where you can decide if this section is going to be restricted from appearing in some specialty store(s). You can also specify default behavior for new specialty stores you might create: Whether the section should only appear in the specialty stores you have

selected here or should it appear in other specialty stores you might create in future if those specialty store's root points to an ancestor of the current section.

5. Choosing Continue leads you to review and create relationship rules for this section. See Building Relationship.

6. After this step you can optionally specify if the section is going to be automatically populated with products based on certain SQL clause as well as specify the order. The auto-placement rule is not currently used. You can also specify Order By clause to specify how the product for a section should be ordered when displayed in the Customer UI. The value for this field could be just one column name from MTL_SYSTEM_ITEMS or it could be comma-separated columns of the same table. Click **Continue**.

7. Now you have created sections and can go and can add products or subsections to this section. Please note that a section can only have a subsection or products as children. You can also add products to a section when you are working on the product (Building product catalog). To add product now, search products by clicking on Add Product that will lead to a pop-up page and then selecting one or more from the result and hitting Add. When done, choose **Done** to close the pop-up and return to the main store manager window.

Guidelines

- The section code is the name used in the template to directly access the section information.
- The display style you choose for products in this section is the style used to display products on a section page for this section.
- You can choose "Include in all future specialty stores if the stores root section is ancestor of this section" for the Included Specialty Stores screen which ties the section to its ancestors. When a specialty store is added to or deleted from the ancestor, the same change applies to all descendant sections.
- The Relationships page displays existing relationships for a section.

Examples

Store Hierarchy:

1. Go to the Hierarchy tab.
2. Click **Add Section** in the right frame to create a new section.
3. Enter Featured Products as the name as well as the code.

4. Select **Featured** as the section type. Everything else is optional.
5. Click **Continue**.
6. On the Multimedia Components screen, everything is optional. Choose **Continue**.
7. Accept the defaults on the Specialty Store and Navigation Relationships screens, and choose **Continue**.
8. The auto-placement rule in the Advanced Settings screen is currently not supported. If required, enter Order By clause. Click **Finish**.
9. In the left frame, click **Refresh**. Expand the Home node, which should have the newly-created section under it.
10. Highlight **Home**, and click **Create**. Repeat the above steps to create another section. Call it Books, section type Navigational.
11. Repeat again for Music, Electronics, Computers, and so on.

Adding Products To Sections:

1. Click on a section name (for example, “Featured Products”) to add products to the section.
2. Click **Add Product** on the children page for the section.
3. Search for a product to add (for example, “Sentinal Multimedia”).
4. Check the selected products, and click **Add**.
5. Click on the product name, and change its posting status to Published.
6. Repeat the above steps to add more items.

Defining Category Mappings

Every product is assigned to a category in Oracle Inventory. Use this procedure to set up defaults for products belonging to the category. Note that in release 11.5.2 you can only specify defaults for categories belonging to a **primary display category** set - the value of IBE:category set profile. For other categories, select store-level defaults.

Prerequisites

- Products must be assigned to categories in Oracle Inventory in order to be returned upon a search of those categories.
- Multimedia, templates, and styles must exist before you can assign them to a category.
- IBE: Category set profile must be set

Steps

1. In the Category tab, conduct a search for categories.

   The Categories page displays a list of item categories from Oracle Inventory which belong to the category set specified in the profile IBE: Category Set along with related templates, display styles, and multimedia components. A

2. Double-click the category name that you want to update.

   The Templates Assigned page lists all template names and default source template files for the chosen category.

3. If you want to add a template, then perform the following steps:

   a. Click **Go.**

      A list of available template names appears.

   b. Select a template.

   c. Click **Add.** The pop-up box closes when you select Done**.**

4. In the Category menu, choose **Display Styles.**

   The Display Styles page lists all display styles you defined in the Setup tab. For each display style you can chose a template from the templates assigned.

5. Select **Update.**

   When in the store at the point at which you wish to display product using a particular display style, Oracle iStore 11*i* will select the appropriate template based upon the following:

   If there is a product-specific template for the given display style, then it is used.

   If the product belongs to a category in **Primary Display Category Set,** then the defaults specified here will be sued. If you do not specify a mapping at the category level, then the store-level default is used.

If the product does not belong to any category in the above category set, then the store level defaults are used.

6. In the Category menu, choose **Multimedia Components.**

   The Multimedia Components page lists all multimedia components you defined in the Setup tab.

7. Optionally, assign multimedia names to multimedia components and click **Update.**

   When the multimedia component is used in a store page that relates to the product category, then the multimedia name retrieves the correct multimedia file according the specialty store the customer is in and the customer's language. A product level multimedia component can override the category level multimedia component. The resolution rule is similar to those in display styles.

8. Go to the Category tab. You should see `Inv.Items` as the category set.

Defining Products

Use this procedure to build your product catalog.

Prerequisites

- Products must exist in Oracle Inventory.
- Display styles must exist before you can assign them to a product. See "Defining Display Styles".
- Hierarchy must be defined before you can assign products to sections. See Defining Product Hierarchy section.

Steps

1. In the Product tab, choose **Products.**

   The Products page appears.

2. Enter search information to find products you want to include in your catalog.

   The Products page lists products in Oracle Inventory that match your search criteria and displays existing product catalog information for those products.

   The search criteria are:

   - Name

- Part number
- Belongs to category
- Created after date
- Created before date
- Status - values should be PUBLISHED or UNPUBLISHED
- New - products created in last x days where x is the profile value of - IBE: Number of Days for New Items
- % can be used for a wild character search

3. Click the product name that you want to edit.

   The Basic Description page displays the inventory description and any descriptions you have already added to the product catalog.

4. If you want to make the product available to be sold in your store, then change the posting status to **Published** and click **Update.** The product is published immediately.

   If the product is not published, you will see the Publish button in the Product Listing page that will take you through the steps required to publish. Alternatively, you can click on **Name** and follow the steps below. If the product is published, then you can modify information as noted below. Note that the information is immediately published. To make changes, first unpublish the product and then republish the product after making the changes.

   If no products are showing up on this screen, it is likely because the WEB_STATUS flag in the MTL_SYSTEM_ITEMS table is NULL. In order to see items in the Merchant UI, they should have a web status of either UNPUBLISHED or PUBLISHED. To make an item show up in the Merchant UI, log in to Oracle Forms, choose the Inventory, Vision Operations (USA) responsibility, and go to **Items > Master Items > Vision Operations**. Choose F11 to enter a search query, then choose Ctrl-F11 to execute it. Use the Web Option tab to publish the desired item(s).

At the time of this writing, a bug in Oracle Inventory is preventing the Web Option tab in the Master Items form from working properly. As a temporary workaround, log into the database via SQL*Plus as apps/apps, and update MTL_SYSTEM_ITEMS directly as follows:

```
UPDATE MTL_SYSTEM_ITEMS
SET WEB_STATUS='PUBLISHED'
WHERE INVENTORY_ITEM_ID=<ID number of the item to be published>
AND ORGANIZATION_ID=<Inventory Organization ID>;
```

Note 1: The value for Inventory Organization ID should be the same as IBE_ITEM_VALIDATION_ORGANIZATION profile.

Note 2: Products in the published status are visible on the store. Be careful about changing information as the changes go to the production system immediately. Development recommends that you unpublish the product, make changes, and then republish the item.

5. Optionally, enter or modify the short and long descriptions and click **Update.**

   The descriptions are saved and are available to display in your store.

6. In the Products menu, choose **Hierarchy Paths.**

   The Hierarchy Paths page displays the hierarchy of sections that have been set up for the store.

7. Optionally, you can remove or add parent sections for the part and edit the dates and display order.

8. In the Products menu, choose **Category and Display Styles.**

   The Category and Display Styles page displays the category to which the product belongs and lists all display styles and any template names already assigned to the product.

9. Choose a template name for each display style that you want to use for the product. Answer **Yes** if you want the category level template used as the default rather than the product-level template for the display style. Click **Update.**

10. In the Products menu, choose **Multimedia Components.**

    The Multimedia Components page lists all multimedia components you defined in the Setup tab.

11. Optionally, assign multimedia names to multimedia components and click **Update.**

12. In the Products menu, choose **Relationships.**

    The Relationships page displays existing relationships between the product and other products or sections and the rules for those relationships, such as the product to show for an upsell or cross sell.

13. If you want to add related items for a relationship, go to "Defining Relationship Rules".

14. In the Product menu, choose **Specialty Store.**

    The Specialty Store page lists the specialty stores where the product will be displayed. By default the product appears in those specialty stores to which the product **parent section** belongs.

15. Select the specialty stores that can display the product and then click **Update.**

Guidelines

- Only products with a status of Published can be sold in your store.

Defining Relationship Rules

Relationships are used for merchandising, for example, to offer a substitute product for a product that is out of stock. To speed your rule building process, use this procedure to relate products, categories, and sections to other products, categories, and sections. One relationship type can contain either rules created using the rule builder or one SQL rule. It cannot contain both.

Using SQL rules to define relationships by querying the database on particular fields is a tactic primarily used by consulting or other highly technical personal. Most store managers will use the mapping rules.

The mapping rules define relationships in a from-to form. The type of from and to objects can be Item Categories (defined in Oracle Inventory), Sections or Hierarchies (defined in Oracle Oracle iStore 11*i*), or Items (defined in Oracle Inventory). For example, if you choose 3 objects in the from list and 2 objects in the to list, a total of six rules are built.

The application evaluates each mapping rule and inserts rows in a table maintains the preevaluated relationships. For example, if you have a category - that has two products assigned - in your From list, and have a section - that has four products - in your To list then Oracle Oracle iStore 11*i* creates a total of eight product relationships.

Business needs drive creation of the relationship rules. Oracle iStore 11*i* ships with the following seeded relationships:

- RELATED
- SUBSTITUTE
- CROSS_SELL
- UP_SELL
- SERVICE
- PREREQUISITE
- COLLATERAL
- SUPERSEDED
- COMPLIMENTARY
- IMPACT
- CONFLICT
- MANDATORY_CHARGE
- OPTIONAL_CHARGE
- PROMOTIONAL_UPGRADE

These seeded relationship types are also seeded in Oracle Inventory for Item Relationships. Unlike in Oracle iStore 11*i*, the user cannot add more relationship types in Inventory. If you use Oracle iStore 11*i*'s Java API to retrieve related items given an item ID and a seeded relationship type, you will get related items defined in Oracle iStore 11*i* plus the ones defined in Inventory.

Relationships

1. Go to the Relationship tab; review the seeded relationships (note that only SUBSTITUTE is used by the store now).
2. Click SUBSTITUTE to create a relationship between items (e.g., item B may be a substitute for item A if item A is out of stock).
3. Click **Create Rule**.
4. In the middle frame, search for the base product, and click [ß] to add it to the From List.
5. Search for the related product, and click the arrow to add it to the To List.

6. Click **Done** to save the relationship.

Prerequisites

Products must exist in Oracle Inventory.

Steps

1. In the Relationship tab, choose **Create.**

   The Create Relationship page appears.

2. Enter relationship information, select **Create.**

   Relationship detail page appears and you can choose to specify the pairs of related items by SQL query or by mapping rules. Choose an option and select **Create Rule.**

   The Create Rules page appears.

3. If you chose **Create a SQL Rule**, go to step 10. If you chose **Create Mapping Rules** in step 2, conduct a search to view products, categories, or sections in the center table.

   The search results appear in the table.

4. Select the items in your search results that you want to be in the *from* side of your rule, and click the left arrow.

   The selected items appear in the From List.

5. Conduct a search to view products, categories, or sections in the center table. This text appears only if you choose mapping rules option in step 2.

   The search results appear in the table.

6. Select the items in your search results that you want to be in the *to* side of your rule, and click the right arrow.

   The selected items appear in the To List.

7. Repeat as needed to complete your *from* and *to* lists for this rule.

8. Click **Done** to submit the relationship rule creation. The **Relationship Detail** page appears. The application generates a rule from every object in the *from* list to every object in the *to* list.

   Or select **Preview Rules** to validate or exclude the relationship rules to be added. The Preview Rules Page appears. At this point the rules have not yet

been added to the system. You may exclude any rules not needed. When finished click **Done** to see the **Relationship Detail** page.

9. From the **Relationship Detail** page, you may select a link for each rule to view all of the product to product relationships generated by that rule.

   Or click **View All Results** to view all of the product to product relationships generated by all of the rules in this relationship type.

   From either of the previous two options, the **Rule Results** page displays the product-level relationship results.

   If you don't want to include one or more of the generated rules, then select **Exclude** and click **Update.**

   The excluded product-level relationships can be re-included.

10. The following incomplete SQL displays:

```
Select msi.inventory_item_id
From mtl_system_items msi
Where
```

    The SQL should return only the column `inventory_item_id` in `mtl_system_items` table. You can add as many tables as you want in the From list and add any conditions in the Where clause.

Pricing Setup Example (Defining Simple Freight Charges)

This section describes an example of how to set up simple fixed-amount freight charges in the store.

1. Go to Oracle Forms, log in as SYSADMIN / SYSADMIN, choose Oracle Pricing Manager responsibility, go to Modifiers à Define Modifier. (If you don't have the Oracle Pricing Manager responsibility, grant it to the SYSADMIN user first.)
2. Click LOV for Type.
3. Click Freight and Special Charge List.
4. Enter some identifier for the modifier in the Number field (e.g. IBEFR01).
5. Enter some description in the Name field (e.g. IBE Freight Modifier).
6. Click in Modifiers Summary Modifier No, and enter a number (e.g. 1).
7. Click in Level, select modifier level (e.g. Line).
8. Click in Modifier Type, select Freight / Special Charge.

9. Enter a Start Date and an optional End Date.
10. Scroll right. Click in Pricing Phases, choose Line Charges from the LOV.
11. Click on the Discounts / Charges tab.
12. Click in Charge Name, select Freight Costs from the LOV.
13. Click **Application Method**, select Amount. Enter a per-line freight cost dollar amount (e.g. 3.00).
14. Save the form.

Setting Up Product Searches

Oracle iStore 11*i*'s product search feature provides merchants with ability to search for products they want to sell.

The product search feature in iStore is implemented using the InterMedia text search utility of Oracle 8i database. The product information (description and long description) is first loaded in an iStore table (IBE_CT_IMEDIA_SEARCH) for the first time via a concurrent program. This step is generally performed after the merchant has loaded his inventory with products. Once the data is loaded, any change to product information is updated in the iStore table through a database trigger call on the inventory table. This keeps product information current in the search table. Once the data is moved in the search table, the iMedia index is created to facilitate search capability of the keywords.

Note: You must ensure that both Oracle Inventory and Oracle InterMedia are installed and configured properly before setting up store search. Refer to the Oracle InterMedia documentation for details on how to set up and configure InterMedia.

What Information do you Store in the Search Table?

The IBE_CT_IMEDIA_SEARCH is basically a denormalized table of MTL_SYSTEM_ITEMS_TL and MTL_ITEM_CATEGORIES.

The core text on which you search for is stored in a clob called INDEXED_SEARCH. Currently it stores a concatenation of name and description of products. The table also stores inventory_item_id, organizationId, category_id, category_set_id and the web status field from MTL_SYSTEM_ITEMS_B table.

What Attributes of a Product can I Search for?

Name and description of a product. They are stored as description and long description columns in MTL_SYSTEM_ITEMS_TL table.

Does Search Have any Dependencies?

Product search needs version 8.1.6 of the database with the InterMedia option correctly installed. It also requires the 11*i* version of the Oracle Inventory schema.

Search Setup Steps

First the merchant sets up his inventory, under a common master org id. Points to remember while setting up inventory include:

- Give products unique names
- Do not leave category names (concatenated segments) blank or non-unique, it can be null or non-unique in the database but will show as blank/multiple times in the categories lov in your customer home page

Make sure the products have the web enabled flag under web options set to PUBLISHED. You can also query this field by examining the web_status column of the item.

The next major step is to run the concurrent program to populate the search table with data and to create the intermedia text index.

Populating the Oracle iStore 11*i* Search Table

1. Log on to Oracle Forms as SYSADMIN/SYSADMIN.
2. Select the **Oracle iStore 11*i* Concurrent Programs Manager** responsibility (as always, if you don't have this responsibility, use the System Administrator responsibility to grant it to yourself).
3. In the pop-up window, choose Single Request, and click OK.
4. Click the LOV button in the Name field, and select **iStore Search Insert**.
5. Click **Submit** to start the concurrent request. Note the request ID.

   You can monitor the progress of your request by looking at the request log and output files in $COMMON_TOP/admin/log/l<request ID>.log and $COMMON_TOP/admin/out/o<request ID>.out, respectively.

   You can also view the request status by selecting View Requests, and searching by the request ID.

Note: You will only be able to search for products whose WEB_STATUS is 'PUBLISHED'.

This process can take substantial amount of time (depending on number of items you have) As an estimate for about 300 thousand items in inventory this program can take about 45 minutes to run.

The concurrent manager will call IBEVCSMV.sql script, which will move the product data from the inventory table to Oracle iStore 11*i*'s search table. When this job is running, the search tables are purged and product search will not work correctly on the store. Since this batch job deletes data from the search table, the rollback segment should be large enough for the process to complete.

Once the request is complete, you will be able to search for products based on name and description. If additional product attributes are to be added in the search, this sql script needs to be modified to add the extra search column.

Setting Up Search Profile Options

iStore search also needs 2 profile options to be setup:

Table 4–6 Search Profile Options

Profile Option	Description
Search Lines Per Page	This option sets the number of lines to be displayed per page. If this profile option is not set, the code default of 20 lines will be used.
Total Search Results	This option sets the maximum cap on the search results. For example, if the user searches for a very common keyword (not in the stop words list), then the search process will stop after the max cap set as per this profile option. If this profile option is not set, then the code default of 500 results will be used.

Creating Search Index Tables

In order to be able to run external procedures to create a search index table, please ensure that ENVS is included in your SID_DESC part of listener.ora as follows:

1. Go to 8.1.6 ORACLE_HOME:

```
cd /u02/visappl
ksh
 . ./APPSORA.env
cd $ORACLE_HOME/../8.1.6/network/admin
```

This directory should contain listener.ora. Verify that listener.ora contains the following:

```
(SID_DESC =
    (SID_NAME = PLSExtProc)
    (ORACLE_HOME = /u04/visora/8.1.6)
    (ENVS = LD_LIBRARY_PATH=/u04/visora/8.1.6/ctx/lib)
    (PROGRAM = extproc)
)
```

Before Creating the search index table, please make sure that the Oracle ConText server is up. Use the following command to check:

```
$ ps -ef | grep ctxsrv
```

If it doesn't exist, start the Oracle ConText server as follows:

1. Go to 8.1.6 ORACLE_HOME:

```
cd /u02/visappl
ksh
 . ./APPSORA.env
 cd $ORACLE_HOME/../8.1.6
 . ./VIS.env
```

This will setup 8.1.6 ORACLE_HOME env. Run the following command:

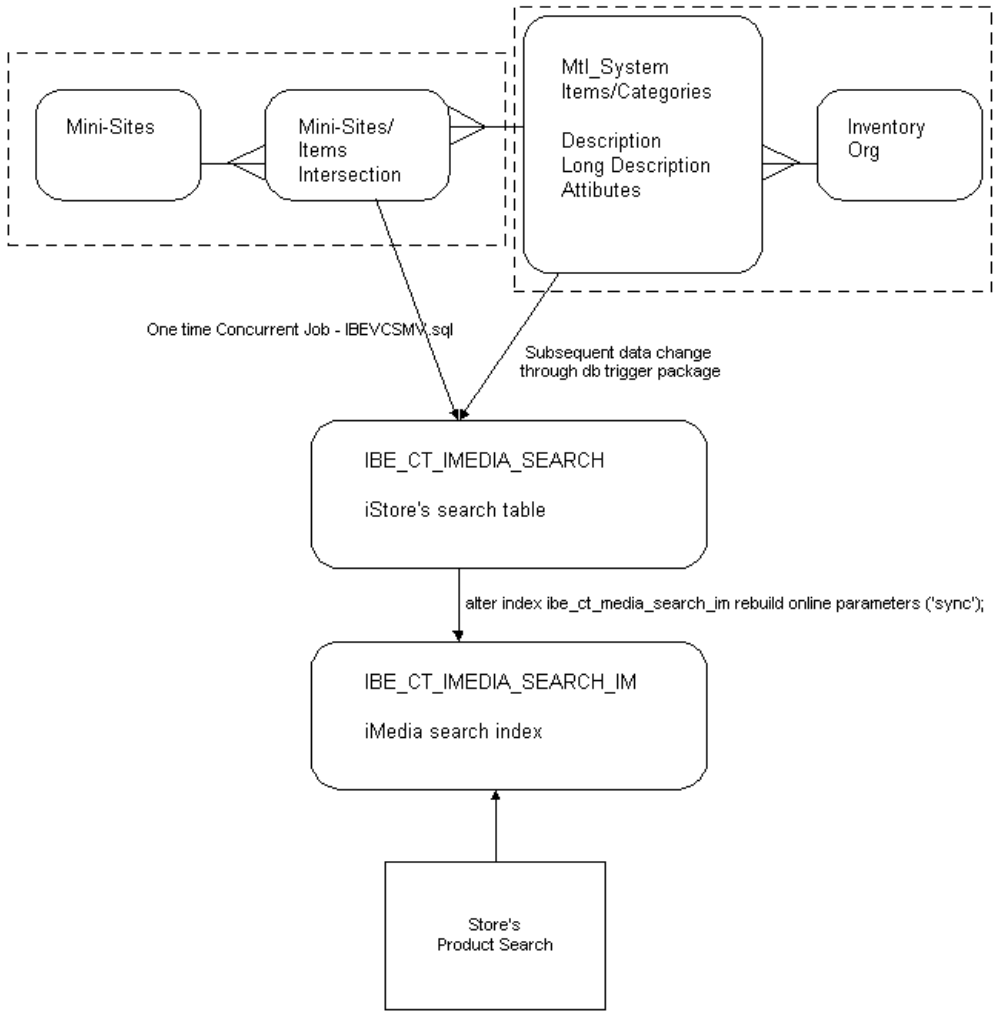
```
ctxsrv -user ctxsys/ctxsys&
```

What are the PL/SQL and Java/ JSP's Involved in Search?

The following program units are used in product search process and can be modified to add other attributes in product search.

- java / catalog / Search.java (main java program that executes the query)
- ibecskr3.jp (Search result jsp)
- ibezmenu.jsp (Main homepage)
- ibecskrf.jsp (Search result jsp)
- IBEVCSMV.sql (one time load of product descriptions through concurrent manager)
- IBEVCSKS.pls (package specification)

- IBEVCSKB.pls (package body)
- IBEVIDTS.pls (package specification for the main database trigger)
- IBEVIDTB.pls (package body for the main database trigger)
- java / catalog / PrdRec.java (definition of search result object)



Customizing Search

If you need to add more attributes of the item to search for, you will need to modify IBEVCSMV.sql for the initial load and the plsql triggers mentioned above to make sure that updates to these attributes get propagated to the search table.

1. Modify the search package (IBEVCSKS.pls and IBEVCSKB.pls) for adding the additional product search attributes. By default, only product name (description column) and the product description (long description column) are included in the search.

   If additional attributes are to be added in the product search, the parameters for the package specification and body will have to be changed accordingly, with the new attributes. This package moves the subsequent changes in the product information, to Oracle iStore 11*i*'s search table. Any insert / delete / update on MTL_ITEM_CATEGORIES, any delete / update on MTL_SYSTEM_ITEMS_B and any delete / update / insert on MTL_SYSTEM_ITEMS_TL table will move the change to the search table through this procedure. This procedure is called from the main database trigger procedures, as explained in the next step.

2. If new parameters are added to the search package in step 1 above, the call to the search package will have to be modified in the main database trigger package IBEVIDTB.plb. This package body calls all the iStore's ERP related db trigger procedures, including the search package procedures. The call to search package will have to be modified to include the new parameters added to the search move procedures.

3. The database trigger on the product tables will call the main db trigger package, as described in the step 2, to move the product data change to the iStore search table.

   Step 2, however, will not recreate the iMedia index to include the changed information in the search table. Administrators must rebuild the iMedia search index every time a new product is added or an existing product is changed / deleted. This can be achieved by recreating the iMedia index IBE_CT_IMEDIA_SEARCH_IM through OEM utility or through executing the command "ALTER INDEX IBE_CT_IMEDIA_SEARCH_IM REBUILD ONLINE PARAMETERS ('sync');" in SQL*Plus. Note that you should have privileges to alter the iMedia index. After this step, the modified product information will be visible in the product search process of iStore.

Adding Stopwords to Searches

There are many search words like "and," "oracle," "if," "then," etc. which are very common and will return numerous search results. Search results may not be relevant to the user's query if such common search keywords are used. In addition, using common keywords in searches uses processing resources and will slow down performance. These common keywords can be excluded from the search by using the "Stop Words" utility in iMedia.

Log into OEM as CTXSYS to see the stop words in the Stop List. Additional stop keywords can be added to the stop list.

iStore Search Query

```
createQueryString(String keywords,
                  String operator,
                  String notKeywords,
                  String categoryId,
                  String maxRowNum,
                  String selectList,
                  String orderByList)

select  i.inventory_item_id, i.description, i.category_id,
    score(100) nearness
from ibe_ct_imedia_search i, mtl_system_items_b b
where contains (i.indexed_search, 'laptop' , 100) > 0
and i.language = userenv('LANG')
and i.category_id = i.category_id
and i.organization_id = 204
and exists (
        select 1
        from jtf_dsp_section_items s, jtf_dsp_msite_sct_items b
        where s.section_item_id = b.section_item_id
        and b.mini_site_id = 10120
        and s.inventory_item_id = i.inventory_item_id
        and (s.end_date_active > sysdate or s.end_date_active is null)
        and s.start_date_active < sysdate
        )
and rownum < 200
and i.inventory_item_id = b.inventory_item_id
and i.organization_id   = b.organization_id
order by SCORE(100)
/
```

Setting Up Credit Card Payments in iStore

Use the following procedure to set up credit card payment functionality in iStore:

1. iStore Setup

   a. Set the IBE: Authorize Payment Offline During Normal Checkout profile option to **No: online authorization**.

   b. Set the Finalize Order On Error in Authorize Payment profile option to the proper value. (This profile will only take effect when authorize payment error is system error.)

      Yes: submit order even if authorize payment error is system error.

      No: catch system error and throw error message.

2. Order Capture Setup

   Set the ASO: Credit Card Authorization profile option to **No** at the iStore application level. (Do not set this profile at responsibility and user level).

3. Accounts Receivable Setup

   Make sure payee id is assigned to receipt method:

   a. Navigate to **Receivables Manager > Setup > Receipt classes.**

   b. Query the receipt method with Name = Credit Card and make sure merchant id field has the payee id.

4. iPayment Setup

   a. Perform the manual post-installation configuration steps described in the latest release of the *Oracle iPayment Implementation Guide*. You should also be familiar with typical iPayment administration operations which are documented in the *iPayment Concepts and Procedures* manual.

   b. In order to integrate iPayment with iStore, the following special steps need to be performed when you create a payee:

      – Go to the iPayment UI Administration screen.

      – Click the **Payee** tab.

      – Click the **Create** button.

        Fill in the form. Pay attention to two important parameters:

Payee Identifier has to be a valid merchant id in AR, otherwise integration will fail. See the Accounts Receivable Setup section of this procedure for details.

CyberCash Id: this an Id acquired after setting up an account with CyberCash.

5. Click the **Credit Card** check box.
6. Click the **Create** button at the end of page to submit the form.

Testing the Store

Test the storefront with the following URL:

http://<host>:<apache port>/html/ibezhome.jsp?minisite=<minisite ID>

where <minisite ID> is the ID of the specialty store created above. Try browsing the catalog and placing an order.

Issue Reporting Guidelines

Use the following guidelines to report problems and file bugs.

Check store functionality in the following order:

1. Check whether all server processes (Oracle, TNS listener, WebDB listener, Forms server, Reports server, Apache, etc.) are up and running.
2. Check JTF login by going to

   http://<host>:<apache port>/html/jtflogin.jsp

   and try to log in as sysadmin/sysadmin. If login fails, the problem is with JTF. File a bug on JTF (product code 481).
3. Check Oracle iStore 11*i* Merchant UI login by going to

   http://<host>:<apache port>/html/jtflogin.jsp

   and try to log in as ibe_admin/manager. If login fails, the problem is with the Merchant UI. File a bug on Oracle iStore 11*i* (product code 384, component SPCLTYSTR).
4. Check the customer UI by going to

   http://<host>:<apache port>/html/ibezhome?minisite=<minisite ID>

and see if the store comes up. If not, file a bug on Oracle iStore 11*i* (product code 384). Use the error message as guidance for which component to specify in the bug.

5. If the store comes up but there are problems adding items to the shopping cart and / or placing orders, use the Oracle Forms UI to check if Order Capture is working. Log in as SYSADMIN / SYSADMIN, select Order Capture Sales Manager responsibility, select Order Capture, and enter a quote. (Refer to Oracle Order Capture documentation for further details.) If OC is not working, file a bug on Oracle Order Capture (product code 769). If OC is working, file a bug on Oracle iStore 11*i* (product code 384, component SHPCRT).

5

Implementing Oracle iMarketing

This chapter describes the implementation of Oracle iMarketing after you have verified your installation and dependency setup. Topics include:

- Overview of Oracle iMarketing
- Documentation Dependencies
- Steps for Implementing iMarketing
- Posting API Details

Overview of Oracle iMarketing

Oracle iMarketing is an internet marketing application designed to increase online revenues, build customer loyalty, and extend company brand presence to other sites.

iMarketing is a tightly integrated add-on module to Oracle iStore and is not being sold stand alone in this release.

iMarkerting works like a personalization engine in the Oracle e-business suite. Merchants can use iMarketing to create and execute marketing campaigns targeted at customer segments on their storefronts.

Oracle iMarketing is integrated with Oracle Marketing Online. It leverages its functionality and shares its schema.

Oracle iMarketing helps merchants compete on the online marketplace. Merchants typically use Oracle iMarketing to

- turn visitors into buyers
- grow each customer to maximum value for the merchant
- build and solidify customer loyalty into a solid, long-lasting competitive advantage

Documentation Dependencies

Oracle iMarketing has the following documentation dependencies for successful implementation. See the latest versions of these documentation for tasks important for iMarketing implementation.

- *Oracle Marketing Online Concepts and Procedures*
- *Oracle iStore Concepts and Procedures*
- *Oracle iMarketing Concepts and Procedures*

Steps for Implementing iMarketing

Complete the following tasks in the order given to implement Oracle iMarketing:

Step 1: Create a CRM Resource

Use the following procedure to create a CRM Resource. When a user logs on, she may have access to only certain functionality depending on the resources assigned to her. This is done for security reasons.

Steps

1. Log on to Self Service Applications.
2. Log on as a SYSADMIN user.
3. From the **File** menu, click **Switch Responsibility.**
4. Scroll down and select **US HRMS Manager**.

   Note: You have to add the responsibility to the sysadmin user if it is not already defined.

5. Expand the **People** tab.
6. Double-click **Enter and Maintain**.
7. Click **No** in the window asking for a date change.

   The Find Person window appears. If you have already created an employee, then you can search here. Otherwise, go to the next step to create an employee.
8. Click **New**. People window appears. Enter the last name, title, type, (select **employee**) and other fields.

9. Click **Assignment** and **Save**.
10. On the **File** menu, click **Switch Responsibility.** Select **CRM Resource Manager.**
11. Expand **Maintain Resources**.
12. Double-click **Import Resources**. Selection Criterion window appears.
13. Click the **Name** field and pick the employee assigned in steps 7 and 8.
14. Click **Search** and then click **OK** in the pop-up window. The selected resource screen appears.
15. Click **Save Resources**.

Step 2: Set Up an iMarketing User

An iMarketing user is someone who has the responsibility to log on to the iMarketing application. This user can perform various procedures, such as setting up campaigns, creating rule sets, etc. necessary for iMarketing functionality.

Prerequisites

Step 1: Create a CRM Resource

Steps

1. Log on to Self Service Applications.
2. Log on as a SYSADMIN user.
3. Select the **System Administrator** responsibility.
4. Go to **Security >User >Define.** Users screen appears.
5. Enter a username and a password.
6. In the Person field, select the CRM Resource that you created.
7. Click in the Responsibility field. Responsibilities window appears. Select **iMarketing Administrator.**
8. Click **Save**.

Table 5–1 Profile Options for a User

Profile Option Name	Value	Description
JTF_PROFILE_DEFAULT_APPLICATION	670	Default application ID (670=iMarketing)
JTF_PROFILE_DEFAULT_CURRENCY	USD	Default currency
JTF_PROFILE_DEFAULT_RESPONSIBILITY	21954	Default responsibility ID (21954=iMarketing Administrator)
JTF_PROFILE_DEFAULT_BLANK_ROWS	3	Number of blank rows on Merchant UI forms (can be set to any integer >0).
JTF_PROFILE_DEFAULT_CSS	jtfucss.css	
JTF_PROFILE_DEFAULT_NUM_ROWS	10	
ICX_LANGUAGE		Should be set to American English
ICX_TERRITORY		Should be set to United States.

Step 3: Create the Discoverer End User Layer (EUL)

Oracle Discoverer is used to create workbooks. Workbooks are used to define segments. An End User layer is a layer containing the definition of business areas. For example, to define iMarketing segments, Marketing Online business area is used.

Prerequisites

Oracle Discoverer has been installed.

Steps

1. Create an EUL user in the database.
2. Login to the database using SQL*PLUS. Login as system / manager@<connect string>. The name of the end user layer owner (database user) is EUL_ <language>, for example, EUL_US.

```
create user eul_us
identified by eul
default tablespace user_data
temporary tablespace temp
```

3. Give the following list of grants:

```
connect apps/apps@<connect string>
grant connect, resource to eul_us;
grant select on FND_USER to eul_us;
grant select on FND_APPLICATION to eul_us;
grant select on FND_USER_RESP_GROUPS to eul_us;
grant select on FND_RESPONSIBILITY_VL to eul_us;
grant select on FND_ORACLE_USERID to eul_us;
grant select on FND_DATA_GROUP_UNITS to eul_us;
grant execute on FND_CLIENT_INFO to eul_us;
connect applsys/apps@<connect string>
grant dba to eul_us;
grant select on FND_PRODUCT_INSTALLATIONS to eul_us;
```

4. In the Discoverer Client, the end user layer has to be created with the /APPS_MODE parameter in the command line as follows:

```
Start--->Run
C:/orant/discvr31/dis31adm.exe /CREATE_EUL /APPS_MODE /CONNECT eul_
us/eul@<connect string>
```

> **Note:** This step creates an icon on your task bar. The title of the icon reports the EUL percentage completed. The icon displays in the system tray and if you move the cursor over the icon, it shows a tool tip (small text window) with current status, for example, Creating EUL Tables, 50% Complete.

5. Login as eul_us/eul@<connect string>. Modify the End User Layer Owner's Indexes by running script BISALTIN.sql. BISALTIN.sql which can be found under $BIS_TOP/admin/sql.

6. Locate the biseul.eex file. It should be in $BIS_TOP/admin/import/US.

7. Start Discoverer from the command line:

```
dis31adm /APPS_MODE /APPS_FNDNAM APPS /CONNECT EUL_US/EUL@<connect string>
```

8. On the **File** menu, click **Import**. Import the biseul.eex file.

9. Click **View** and select **Validate**. You should not see any errors. If you see an error, one or more of the previous steps were not successful.

10. Refresh business areas. Run the script BISEULBA.sql. Connect as eul_us/eul@<connect string>. BISEULBA.sql, which should be in $BIS_TOP/patch/115/sql.
11. Click **Tools > Security**. Ensure that the Marketing Online business area is selected. Assign the following roles to the iMarketing user.
    - Oracle Marketing Administrator
    - Oracle Marketing Super User
12. Click **Tools > Register PL/SQL functions**. Click **Import**. This takes a couple of minutes.
13. Select APPS.AMS_DISCOVERERSQL_PVT.EULTRIGGER@POST_SAVE_DOCUMENT and click **OK**.
14. Follow the steps to register a form function. (See Step 4: Register Form Functions). Now you can create a workbook.

Step 4: Register Form Functions

A SQL statement is generated while creating a workbook. This SQL statement has to be stored for using the different functionality in the iMarketing application. Registering the Form Function stores the SQL statement.

Prerequisites

None

Steps

1. Log on to Oracle Applications as Sysadmin.
2. Select the **Application Developer** responsibility.
3. Click the **Application** tab.
4. Click **Function**. Form Functions window appears. Query to check if "AMS_LAUNCH_DISCO" has been registered. If it has been registered, exit Oracle Applications. If it has not been registered, continue with the following steps.
5. Enter AMS_LAUNCH_DISCO in Function and User Function name fields.
6. Enter www in the Type field.
7. Enter Web Discoverer in Description.

8. Click **Web HTML** tab. In HTML call corresponding to AMS_LAUNCH_DISCO function, enter OracleOasis. Run Discoverer.
9. Click **Save** and exit Oracle Applications.

Step 5: Create a Workbook in Discoverer

Creating a workbook is the first step in creating a segment. A workbook defines a segment in terms of a business area. Oracle Discoverer generates a SQL query, which is used to generate segment information.

Prerequisites

Access to the Discoverer Client.

Steps

1. Log on to Discoverer Admin Console (e.g, Dis31adm).
2. Click **Start** on the Welcome window. Enter the following information:
   - username: e.g, eul_us
   - password: e.g, eul
   - Database/connect string: e.g, dom1151
3. Login. The Load Wizard window appears.
4. Click **Open and Existing Business Area**.
5. Scroll down and select **Marketing Online Business Area**.

   Note: The list is not displayed in an alphabetical order.

6. Click **Finish**. Marketing Online business area is displayed with an Administration Tasklist window on the bottom right corner. You may close this window.
7. On the **Tools** menu, click **Security**. Security window appears.
8. For the initial setup, in the Selected Users/Roles secondary window, make sure that the iMarketing App User and iMarketing Administrator responsibility appear as you scroll down. Also observe the symbol associated with the above names, representing user and roles respectively. If you find them, click **Cancel**

and exit the Discoverer from the **File** menu. Otherwise, assign the user and role and click **Apply** and exit Discoverer.

9. Log on to the Discoverer User Console (e.g, Dis31usr). Click **Start**.
10. Login using the same information as in step 2.
11. Scroll to the right most tab or until you see the EUL tab. Click the **EUL** tab.
12. Select EUL_US as the default EUL. Click **OK**.
13. On the **File** menu, click **Connect to Database**. The Workbook Wizard appears.
14. Click **Create a New Workbook**. Use either a table or any other format to display the results.
15. Select the format that you used for displaying the results and click **Next**.
16. Construct the Query by clicking on the items of interest such as Party_id, name, zipcode etc.

    Note: Party_id is a mandatory field.

17. Save the workbook.

    Note: You can save your workbook in the database or on your local drive. Saving on the database allows you to attach the workbook to segments. Saving on your computer lets you reload the workbook whenever the database is refreshed so that you don't have to recreate the workbook.

18. Login to iMarketing.
19. Click the **Audience** tab. Segments screen appears.
20. Click **Create**. Create a segment and click **Go** in workbook field. You will see the work book that you created. If you don't see the workbook, you may not have saved it to the database.
21. Select the workbook and click **Update** in the Create Segment screen to see the SQL query in SQL viewer field.

    After you create a workbook and attach a segment, you need to run the Refresh Party Market Segment concurrent program to update the segment and party association.

Step 6: Create Segments for Anonymous iStore Users

An unregistered user browsing the iStore is known as an Anonymous user. A segment for an anonymous user is created to promote products and / or offers for such a user.

Prerequisites

None

Steps

1. Log on to the database using SQL*PLUS.
   Username: apps
   Password: apps@<connect string>
2. Enter the following statement to get the anonymous user id:

   ```
   Select user_id from FND_USER where user_name='IBEGUEST';
   ```

3. The result retrieved is, for example, USER_ID / 1000004.
4. Use this USER_ID, e.g, 1000004, to create an anonymous segment.

Note: The condition part of the anonymous segment query should look like ".....where.....party_id=1000004;"

Step 7: Run a Refresh Party Market Segment Program

After creating a segment, you have to associate the segment and the users who qualify for the segment. By Running the Refresh Party Market Segment Program, you can associate the users to the segments.

Steps

1. Log on to Self Service Apps.
2. Select **Oracle Marketing Admin App User** responsibility and click **OK**. The Navigator window appears with the setup highlighted in blue.
3. Double-click the **Setup** link.
4. Double-click **Concurrents Requests**. Find Requests window appears.
5. Click **Submit a New Request.**

6. Single request will be selected by default on the newly appeared window. Click **OK**. Submit Request window appears.
7. Click in the Name field and get a list of names of the concurrent programs.
8. Select **Refresh Party Market Segments** program and click **OK** to run the concurrent program. Parameters window appears. Enter the Cell_id.
9. The cell_id, is the same as the segment id that you created. Enter the cell_id and click **OK**. If you do not know this cell_id, click **OK** to update all the segments. This may take a while.

   Note: To find the cell id, navigate to the Audience tab in the iMarketing application and see the URL to get the segment id (cell_id).

10. Watch the status of your request for completion.

Posting API Details

The posting call is included in an iStore JSP page. The method is used for posting content. Posting content might include an image/banner and/or some text which is displayed in a posting.

Posting Calls from iStore Templates

Postings are called from the following iStore templates:

- Product Detail page: ibecitmd.jsp*
- Section page: ibeclfsm.jsp*, ibeclfss.jsp*
- Shopping Cart page: ibescdch.jsp*
- Home Page: ibezhome.jsp*

The following code sample is an example of the JSP code used to call a posting from the iStore template:

```
<%    if (IBEUtil.iMarketingInstalled()) {
        //------------ set parameters for iMarketing: postingId,
numRequested, random
        Integer postingId = new Integer(10160);
        if (postingId != null)
          pageContext.setAttribute("postingID", postingId,PageContext.REQUEST_
SCOPE);
        Boolean random = new Boolean(false);
        pageContext.setAttribute("random", random,PageContext.REQUEST_SCOPE);
        try {
%>
          <jsp:include page="ibapstng.jsp" flush="true" /><br>
<%      } catch (Throwable e) {
        IBEUtil.log("ibezhome.jsp", "iMarketing error", Logger.ERROR);
        }
      }
%>
```

Note: Edit the PostingID in the iStore template(s) to show the desired posting.

Displaying a Posting on an Affiliate Site

The method signature includes two extra parameters in the case of an affiliate page. The call would be embedded in a JSP page which is called from the affiliate site. This would be an HTTP call with the following syntax:

```
http://<remote host name>:<port no>/<root directory>/<iMarketing Class File
Dir>/ibapstng.jsp
```

6

Diagnostics and Troubleshooting

This section contains instructions on error corrections and workarounds that you may encounter with configuring or administering Oracle iStore 11*i*. Topics include:

- Java Applet Warning Workaround
- Error RA-29868 While Executing amviccn.sql
- Search Errors
- PostSales Errors
- Catalog and Pricing Errors
- Shopping List Errors
- Display Manager Errors
- Potential Issues Installing Oracle8i interMedia Text Version 8.1.5

Java Applet Warning Workaround

Use the following procedure to remove the yellow bar displaying a java applet warning.

1. Uninstall JInitiator from your client machine: in Windows 95/98/NT, go to **Start > Settings > Control Panel > Add/Remove Programs**. Find Oracle JInitiator 1.1.7.27 Export, and click **Add/Remove**. Make sure JInitiator is removed successfully.
2. Delete the identitydb.obj file from your client machine (it is usually in the parent directory of your JInitiator installation, i.e., C:\Program Files\Oracle\).
3. From your client machine, FTP to the server.

   ```
   cd <COMMON_TOP>/html
   ```

   where <COMMON_TOP> is the actual value (i.e. /u04/viscomn). You cannot use environment variables in FTP. Make sure you are using binary transfer mode: bin.
4. Download oajinit.exe to your client machine:

   ```
   get oajinit.exe
   ```
5. Change to <APPL_TOP>/admin, where <APPL_TOP> is the actual value (i.e. /u02/visappl).
6. Download the certificate file appltop.cer:

   ```
   get appltop.cer
   ```
7. Close the FTP session.
8. On your client machine, launch a command prompt session, and execute oajinit.exe from the command line. Wait until JInitiator is successfully installed.
9. Change to the directory in which JInitiator is installed (typically C:\Program Files\Oracle\Oracle JInitiator 1.1.7.27 Export).
10. Change to the bin subdirectory.
11. Copy the appltop.cer file into the bin directory.
12. Execute the following command exactly as shown:

    ```
    ./javakey.exe ic appltop appltop.cer
    ```
13. If your browser was open, close and restart it.

14. Log on to Oracle Forms. The yellow bar should not appear.

Error RA-29868 While Executing amviccn.sql

When running the database driver and executing amviccn.sql, you may receive the following error message:

```
RA-29868: cannot issue DDL on a domain index marked as LOADING.
```

This error causes adpatch to halt.

To correct this error, use the following procedure:

1. Open another window or telnet session.
2. Set up the environment as usual.
3. Change to the $APPL_TOP/patch/1306413/amv/patch/115/sql directory.
4. Edit the file amviccn.sql to change the line

   ```
   dbms_sql.parse(curs, 'DROP INDEX '||dropIndex.index_name, dbms_sql.native);
   ```

   to

   ```
   dbms_sql.parse(curs, 'DROP INDEX '||dropIndex.index_name||' FORCE',
   dbms_sql.native);
   ```

   That is, add the FORCE clause to the DROP INDEX command.
5. Save the file amviccn.sql.
6. Use the adctrl utility to restart the failed worker by performing the following procedure:
   a. Execute adctrl.
   b. Accept defaults for all prompts until you get to the Main Menu.
   c. Choose **1. Show Worker Status** to identify the failed worker.
   d. Press **Return** to continue.
   e. Choose **2. Tell Worker To Restart A Failed Job** and enter the ID number of the failed worker.
   f. Choose **1. Show Worker Status** again to monitor the status of the job.
7. If all else fails, stop and restart adpatch.

Search Errors

To troubleshoot Search errors, use the following checkpoints:

- Check the Search log file by executing ctx_output.start_log('log').
- Check that Concurrent Manager is up and running.
- If searches are inaccurate, verify that interMedia was set up correctly. Refer to "Potential Issues Installing Oracle8i interMedia Text Version 8.1.5".
- Look for data stored in the indexed search column.
- If you add multiple items that do not appear in the search table, re-run the Concurrent Manager program Store Search Insert. This program populates the search table in Oracle iStore 11*i* with product information from the inventory tables.

  Note: The search function goes offline while Store Search Insert is running, which takes about forty-five minutes.

- Verify that the listener.ora and tnsnames entries are correct so that the callout to the .dll can be made. For UNIX, refer to "Potential Issues Installing Oracle8i interMedia Text Version 8.1.5".

  For Windows NT, refer to the following example.

```
listener.ora
*********************
LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS_LIST =
        (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC0))
      )
      (ADDRESS_LIST =
        (ADDRESS = (PROTOCOL = TCP)(HOST = sthattil-pc)(PORT = 1521))
      )
    )
    (DESCRIPTION =
      (PROTOCOL_STACK =
        (PRESENTATION = GIOP)
        (SESSION = RAW)
      )
      (ADDRESS = (PROTOCOL = TCP)(HOST = sthattil-pc)(PORT = 2481))
```

```
    )
  )

SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME = PLSExtProc)
      (ORACLE_HOME = E:\Oracle\Ora81)
      (PROGRAM = extproc)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = ORCL)
      (ORACLE_HOME = E:\oracle\ora81)
      (SID_NAME = ORCL)
    )
  )

*****************************************
tnsnames.ora
*****************************************
EXTPROC_CONNECTION_DATA.US.ORACLE.COM =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC0))
    )
    (CONNECT_DATA =
      (SID = PLSExtProc)
      (PRESENTATION = RO)
    )
  )
```

If the code is correct, refresh the dr$libx so it can find the .dll to create the index (my oractxx8.dll is in my $ORACLE_HOME\bin directory). The command to recreate the library is:

```
create or replace library dr$libx as 'e:\oracle\ora81\bin\oractxx8.dll';
```

Start the ctxsrv and re-index.

PostSales Errors

To fix issues with PostSales, first verify that the following prerequisites were done.

- All the views in PostSales are “VALID” in the database.
- Regions exist in Apps. Use the developer responsibility Apps for Web Manager / AK.
- Shipments, invoices and payments were created by the merchant through ERP applications before trying to view them in Oracle iStore 11*i*.

Shipment Page Errors

If errors appear on the Shipment Page, use Patch # 1302329 or IBE Patch # 1331176.

Order Summary Page Records Out of Sequence

The workaround is to use Search to locate an order.

Catalog and Pricing Errors

To troubleshoot Catalog and Pricing errors, use the checkpoints or the following fixes for specific error messages.

Checkpoints

- View the JSP source in the web browser and look for a stack trace.
- Check the log file for a stack trace.
- If you know the JSP or Java method in which the error occurred, search the log file for debug statements from the JSP or method.

Specific Error Messages

Use the following fixes and workarounds for the error messages listed below.

No Items in the Catalog

- Check the following profiles:
  - ASO: Product Organization
  - IBE: Item Validation Organization

- Check the Item setup:
  - web_status='PUBLISHED'
  - start_date_active is NULL or <=SYSDATE
  - end_date_active is NULL or >=SYSDATE

Items Do Not Have Prices

- Check the following profiles:
  - IBE: Pricing Event -- Before Shopping Cart
  - IBE: Request Type to get a Price
- Confirm Items are in the minisite price list.
- Check the log file for the pricing API that is being called (Item.getListAndBestPrices).
- Status Code from Pricing Engine
  - UPDATED
  - DUPLICATE_PRICE_LIST
  - INVALID_UOM
  - IPL: Invalid price list ID, that is, a non-existent price list.

"Add to Cart" Buttons Do Not Appear

"Add to Cart" buttons are not displayed for items that are not orderable on the web and for items that do not have defined prices.

- Check that the item setup is orderable_on_web_flag = 'Y'
- Check the price setup

Shopping List Errors

To troubleshoot Shopping List, perform the following checks:

- Run ctx_output.start_log("log")
- Check the log file for errors in the program flow.
- Check whether data is stored or modified in the database table.
- Run SQL from the log files.
- Run PL/SQL scripts if a problem has occurred in the PL/SQL layer.

Display Manager Errors

If a stack trace or error message indicates that a Display Manager API caused an error, check the top line of the exception Java stack trace for the last class and method that was called and caused the error.

Display Manager uses two sets of log files:

- FWSYS, system log files that are used by JTF and are not used by Applications.
- ibe.log.run, the Application log files that contain the sequence of actions recorded by Oracle iStore 11*i*.

Display Manager Error Messages

Use the following fixes and workarounds for the Display Manager error messages listed below.

TemplateNotFoundException or MediaNotFoundException

This error occurs when either DisplayManager.getURL() or DisplayManager.getTemplate(accessName), DisplayManager.getMedia(accessName)methods or used.

- Verify that the template or media was created in the Admin Console by searching for it by access name in the Admin Console.
- Reboot the server.
- Check the log file for diagnostic messages.

NullPointerException (Template or Multimedia Object is Null)

This error occurs when methods Item.getTemplateFileName() or Section.getTemplateFileName() methods or Display Manager.getSectionMedia(...) are made.

This error indicates that a product or section mapping to a template or multimedia was not defined and that a store level default was also not defined.

- Verify that the mapping for the product or section was specified in the admin console.
- If the category default is in use, check whether a mapping has been defined at the category level for the requested display style.
- Verify that the specified display styles / multimedia components are actually available.
- Verify that the server was bounced after the mapping was created.
- Check the log file for diagnostic messages.

New Template (or Multimedia) Has Been Associated with a Product/Category/Section, But Won't Show Up at Runtime.

- Verify that the association was created successfully in the admin console.
- Verify that the specified display styles or multimedia components are available.
- Verify that the server was bounced after the association was created.
- Check the log file for diagnostic messages.

Potential Issues Installing Oracle8i interMedia Text Version 8.1.5

Use the following procedures to troubleshoot problems installing Oracle8i interMedia Text Version 8.1.5.

Resolving Upgrade Errors

Users may encounter errors when upgrading from versions prior to version 8.1. When that happens, users often install interMedia Text from the Universal Installer and the database assistant does not get started. Then they usually try to install the database objects manually and run into problems.

To resolve this problem, use the following procedure.

Steps

1. The interMedia Text install consists of a SOFTWARE install and a DATABASE OBJECTS install. Install the software using the Universal Installer.
2. Start up the Database Assistant ($ORACLE_HOME/bin/dbassist) to install the database objects. If you cannot install the software objects cleanly using the Database Assistant, review the instructions in Oracle8i interMedia Text 8.1.5.

Note: where the bulletin says "$ORACLE_HOME/ctx/lib/libctxx8.so" you should in fact type in the expansion of $ORACLE_HOME.

Manually Installing ctxsys Data Dictionary

Data Dictionary InstallationinterMedia Text is integrated with the Oracle Database Creation Assistant (DBCA) so the ctxsys data dictionary should be installed when using this tool. If the ctxsys data dictionary does not install, use the following procedure to install it manually.

Prerequisites

- The interMedia Text files are installed.
- The database does not have a ctxsys user.
- The current directory is ?/ctx/admin.
- You can sqlplus internal.

Steps

1. Create the ctxsys user and pass it the ctxsys password, default tablespace, and temporary tablespace as arguments.

   ```
   sqlplus internal @dr0csys <password> <def_tblspc> <tmp_tblspc>
   ```

2. Install the data dictionary:

   ```
   sqlplus ctxsys/<password> @dr0inst <ctxx_library>
   ```

   The argument is the full path to the ctxx library, for instance:

   ```
   sqlplus ctxsys/<password> @dr0inst $ORACLE_HOME/ctx/lib/libctxx8.so
   ```

3. Install appropriate language-specific default preferences. There are more than forty scripts in ?/ctx/admin/defaults that create language-specific default preferences. They are named in the form drdefXX.sql, where XX is the language code (from the Server Reference Manual).

   For instance, to install the US defaults:

   ```
   sqlplus ctxsys/<password> @defaults/drdefus.sql
   ```

   interMedia Text should now be installed and working.

Post-Installation Setup

If this database was an existing ConText site, make sure to remove text_enable from the init.ora. It is no longer used in 8i, and will actually prevent 8i from operating properly. You will get errors such as “Cannot find package DR_REWRITE.”

Ensure that the Net8 listener is running and is configured to invoke external procedures. A brief description of the process is below, and complete details are in the Oracle8i Server Administrator's Guide.

1. Add an entry to the tnsnames.ora:

   ```
   extproc_connection_data =
     (DESCRIPTION =
        (ADDRESS = (PROTOCOL = ipc)
                   (KEY = DBSID))
        (CONNECT_DATA = (SID = ep_agt1)))
   DBSID is the database SID.  ep_agt1 can be named anything.
   extproc_connection_data should not be changed.
   ```

2. Add the following to the listener SID_LIST:

```
SID_DESC = (SID_NAME = ep_agt1)
           (ORACLE_HOME = /oracle)
           (ENVS = LD_LIBRARY_PATH=/oracle/ctx/lib)
           (PROGRAM = extproc))
```

   ep_agt1 matches the CONNECT_DATA SID for extproc_connection_data in the tnsnames.ora. The PROGRAM section tells the Net8 listener to start the external procedure process. The ENVS section, which is shown here for UNIX, will ensure that the environment includes ?/ctx/lib in LD_LIBRARY_PATH. This is needed so that indexing can use the INSO filters.

3. Since the extproc_connection_data ADDRESS section specifies ipc, make sure that the ADDRESS_LIST of listener.ora accepts ipc connections.

   A quick way to test the Net8 configuration is to do:

```
exec ctx_output.start_log('log')
```

   from SQL*Plus. If the setup was not performed correctly, you get the error,

```
DRG-50704: Net8 listener is not running or cannot start external procedures.
```

   To troubleshoot this error, check the following possible causes:

- listener is not running.
- listener.ora is not configured for extproc.
- tnsnames.ora is not configured for extproc.
- listener does not accept ipc connections.

7

Profile Options, Accounts, and Forms Settings

This chapter describes profile option settings, account setups, and Oracle Forms settings that are required for successful implementation. Topics include:

Before You Begin

Before making Oracle Forms settings, ensure that all Oracle Applications server processes are up and running. In particular, if you stopped concurrent managers before applying Oracle Applications patchsets, restart them now by changing to $COMMON_TOP/admin/scripts, and executing adcmctl.sh <APPS username/APPS password> start.

Setting Up JTF Properties

Use the following procedure to set up JTF profiles.

Steps

1. Go to http://<host>:<apache port>/html/jtflogin.jsp, then log in as sysadmin/sysadmin.
2. Click **Advanced** > **Properties**, and select **View JTF** to view JTF properties.
3. Click **Next** to go to the next page. Make sure guest_partyname is `IBEGUEST`, guest_password is `ibeguest2000`, and guest_username is `ibeguest`. To change the password, click on it.

   Note: If you change the password here, it will not be reflected in Oracle Forms. To maintain consistency you must also change the IBEGUEST password manually in Forms.

4. Find and click on **framework.Logging.system.level**, and set sequence 1 value to **debug**.
5. Click **Update** twice.
6. Find service.Logging.common.level, and change it to **debug**.
7. Go to **Registration** > **Default Responsibility**.
8. Choose **Business Users** from the pull-down menu and set the following:
   - Set Default Application ID to 671 (IBE)
   - Set Default Responsibility to 22372 (IBE_CUSTOMER).
9. Choose **End Users** from the pull-down menu and set the following:
   - Set Default Application ID to 671 (IBE)

- Set Default Responsibility ID to 22372 (IBE_CUSTOMER).

10. Click **Submit**.

    You do not receive feedback on the update, and remain on the same web page.

Setting Foundation (JTF) Profile Options

These profiles are required to be set before the Merchant UI can be brought up. The values of these profiles set the basic JTF foundation default elements and values. These profiles are seeded in the Profiles form in ERP and the values are defined by the user (System Administrator). These profiles are set on the user level for IBE_ ADMIN user. The IBE_ADMIN user is seeded with the product and is the owner of these profile options.

Steps

1. Log on to Oracle Forms by opening http://<host>:<web port>/, then clicking on **Apps Logon Links > VIS Logon** through the Forms cartridge (UNIX).
2. Log in as SYSADMIN / SYSADMIN.
3. Choose the System Administrator responsibility.
4. Go to **Profile > System > Find System Profile Values**.
5. Check **Application**, and enter **iStore**.
6. In the Profile field, enter **JTF_PROFILE%**, and click **Find**.
7. Verify and / or set the following JTF profile options at both the site and the application level for iStore:

Table 7–1 JTF Profile Options

Profile Option Name	Value	Description
JTF_PROFILE_DEFAULT_APPLICATION	671	Default application ID (671=iStore).
JTF_PROFILE_DEFAULT_BLANK_ROWS	3	Number of blank rows on Merchant UI forms (can be set to any integer >0).
JTF_PROFILE_DEFAULT_CSS	jtfucss.css	
JTF_PROFILE_DEFAULT_CURRENCY	USD	Default currency
JTF_PROFILE_DEFAULT_NUM_ROWS	10	

8. On the same screen, verify and / or set the following profile option at the application level only for Oracle iStore 11*i*:

Table 7–2 JTF_PROFILE_DEFAULT_RESPONSIBILITY Option

Profile Option Name	Value	Description
JTF_PROFILE_DEFAULT_RESPONSIBILITY	21819	Default responsibility ID (21819=IBE_ADMINISTRATOR).

9. Go to **Profile > System > Find System Profile Values**.
10. Check **Application** and **User**, and enter **iStore** and **IBEGUEST**, respectively.
11. In the Profile field, enter **JTF_PROFILE%**, and click **Find**.
12. Verify and / or set the following JTF profile options at the *user* level for IBEGUEST:

Table 7–3 User-Level IBEGUEST JTF Profile Options

Profile Option Name	Value	Description
JTF_PROFILE_DEFAULT_APPLICATION	671	Default application ID (671=iStore)
JTF_PROFILE_DEFAULT_CURRENCY	USD	Default currency
JTF_PROFILE_DEFAULT_RESPONSIBILITY	22372	Default responsibility ID (22372=IBE_CUSTOMER)

13. Go to **Profile > System > Find System Profile Values**.
14. Check **Application** and **User**, and enter **iStore** and **IBE_ADMIN**, respectively.
15. In the Profile field, enter **JTF_PROFILE%**, and click **Find**.
16. Verify and / or set the following JTF profile options at the user level for IBE_ ADMIN:

Table 7–4 IBE_ADMIN JTF Profile Options

Profile Option Name	Value	Description
JTF_PROFILE_DEFAULT_APPLICATION	671	Default application ID (671=iStore)
JTF_PROFILE_DEFAULT_CURRENCY	USD	Default currency

Table 7–4 IBE_ADMIN JTF Profile Options

Profile Option Name	Value	Description
JTF_PROFILE_DEFAULT_RESPONSIBILITY	21819	Default responsibility ID (21819=IBE_ ADMINISTRATOR)

Setting IBE Profile Options

The IBE profile options and values define the way a customer and merchant UI will work. The profiles set is seeded in the product and displayed when the administrator sets the values for these options.

Steps

1. Log on to Oracle Forms by opening http://<host>:<web port>/, then clicking on **Apps Logon Links > VIS Logon** through the Forms cartridge (UNIX).
2. Log in as SYSADMIN/SYSADMIN.
3. Choose the System Administrator responsibility.
4. Go to **Profile > System > Find System Profile Values**.
5. Check **Application**, and enter **iStore**.
6. In the Profile field, enter **IBE%**, and click **Find**.
7. Verify and/or set the following iStore (IBE) profile options at the application level:

Table 7–5 IBE Profile Options

Profile Option Name	Recommended Values	Description
IBE: Authorize Payment Offline during normal Checkout	Yes	This option is for allowing the checkout and payment authorization offline. If the payment system is set to perform online validation and returns a system error (rather than an authorization error), setting the profile to Yes enables the order to be taken and sent to Order Management for offline authorization. Otherwise, the user is shown an error message.
IBE: Cache	Yes	Specifies whether to enable the store cache for sections and items
IBE: Category Set	Inv.Items	The category set id, after creating the categories for the items. The category for determining the display category for an item. If the item belongs to a category in this category set, then it will use the default display style mappings for this category. Otherwise, the default mappings at the store level are used. The value “Inv. Items” is for the Vision database.
IBE: Default Contract Template Name	<your template name>	Specifies the default Oracle Contracts template name for iStore. Used only if Oracle Contracts is installed and configured.

Table 7–5 IBE Profile Options (Cont.)

Profile Option Name	Recommended Values	Description
IBE: Default Display Style	Catalog Featured Item Style	While creating the display styles on the merchant UI, one of the display styles is to be set to as default. That is, if no mapping is successful for a given template the default display style will be used. This profile should be set after creating at least one display style.
IBE: Default Item Media	(Leave blank)	Specifies the default image which will be displayed if the mapping at the item / Section / Store level fails to find a multimedia for the item.
IBE: Default Item Template	(Leave blank)	Specifies the name of the default item template. Create the template first in Merchant UI and then select from LOV list.
IBE: Default Order Admin Email	<email address of party to be notified>	Specifies the default order admin email address
IBE: Default Payment Term	IMMEDIATE	Specifies the default payment term
IBE: Default Request Terms Change Email	<email address of party to be notified>	Specifies the default Request Terms Change email address
IBE: Default Section Media	(Leave blank)	Name of default multimedia file for sections
IBE: Default Section Template	(Leave blank)	Name of the default template for sections

Table 7–5 IBE Profile Options (Cont.)

Profile Option Name	Recommended Values	Description
IBE: Enable Preloading of Cache for Catalog	Yes	This option will pre-load the catalog, depending on the size specified in the cache limits profile, when the first user logs in. At the expense of the first hit, the subsequent catalog search and navigation becomes faster due to this option set to Yes.
IBE: Finalize Order On Error in Authorize Payment	No	If Yes, allows orders to go through even if payment authorization fails
IBE: Item Cache Size	200	Specifies the maximum number of items to cache on the middle tier
IBE: Item Validation Organization	Inventory Master Organization	For vision demo, master organization is Vision Operations (204).
IBE: Items Per Page for Display	20	Specifies the number of items to display on a single page
IBE: Items Per Section for Display	(Leave blank)	Specifies maximum number of items to display per section. If no value is entered, all items in a section will be displayed.
IBE: Line Level Pricing Event for Shopping Cart	Enter Order Line	Specifies the type of pricing event to be used at the line level in the shopping cart
IBE: Merge Shopping Cart Lines	Yes	Specifies whether you want to merge shopping cart lines
IBE: No of Results in Search	200	Specifies the number of hits returned by a store search
IBE: Notification User Role	System Administrator	User responsibility for setting notifications
IBE: Number of Days for New Item Definition	5	An item created within this many days is considered a new item in the merchant UI

Table 7–5 IBE Profile Options (Cont.)

Profile Option Name	Recommended Values	Description
IBE: Number of Invoice / Order Lines displayed	10	Specifies the number of invoice / order lines displayed in Order Tracker
IBE: Number of Menu Subtabs	10	Specifies the number of menu subtabs
IBE: Number of Menu Tabs	5	Specifies the number of menu tabs
IBE: One Click Consolidation Time	10	Specifies the time interval, in minutes, in which the one click shopping cart will be converted in an order by the concurrent batch job
IBE: One Click Error Behavior	SAME PAGE	Displays one-click error messages on the same page they occurred
IBE: Order Tracker Object Cache	Yes	Specifies whether order tracker Java objects are cached in the middle tier (recommended).
IBE: Preferred Support Level for Shopping Cart	(Leave blank)	Specifies the preferred support level for end user shopping carts.
IBE: Pricing Event before Shopping Cart	Enter Order Line	Specifies the user-defined pricing event (defined in Adv. Pricing) for the catalog stage
IBE: Pricing Event for Shopping Cart	Enter Order Line	The merchant can choose any user-defined pricing event for processing the price for the shopping cart. These user-defined pricing events should be first created in QP.
IBE: Request Type to get a Price	Order Management Order	Specifies the application name for the source requesting the price from the pricing engine
IBE: Reserve Items On Every One Click	No	Not used in the current release

Table 7–5 IBE Profile Options (Cont.)

Profile Option Name	Recommended Values	Description
IBE: Retrieve Price When Displaying Items	Yes	If Yes, retrieve prices for an item's primary UOM based on the specialty store's price list id when loading the item. Otherwise, prices will be retrieved when the price APIs are called.
IBE: Search Lines Per Page	20	Specifies the number of search results displayed on a single page
IBE: Section Cache Size	20	Specifies the maximum number of sections to cache on the middle tier
IBE: Sections Per Page for display	10	Specifies the number of sections to be displayed on one catalog page
IBE: Shopping Cart Price based on Owner	Yes	If set to Yes, the shopping cart price will be based on the shopping cart owner if retrieved by someone sharing the cart. Otherwise, the cart price will be recalculated based on modifications to the cart made by the person sharing the cart.
IBE: Use Aol Menu	No	Specifies whether to use AOL's menu framework
IBE: Use Catalog exclusions	Yes	Specifies whether to use catalog exclusions. If exclusions are never specified in any specialty store, then set to No to improve performance.
IBE: Use Category Search	Yes (required)	Used to enable / disable category search. Only the Yes value is currently supported.
IBE: Use Express Checkout	(Leave blank)	Used to enable / disable express checkout.

Table 7–5 IBE Profile Options (Cont.)

Profile Option Name	Recommended Values	Description
IBE: Use Line Level Shipping	No	Used to enable / disable line level shipping
IBE: Use Section Bin	Yes	Used to enable / disable section bin
IBE: Use Section Path	Yes	Used to enable / disable section path
IBE: Use Shipping Instructions	No	Used to enable / disable shipping instructions
IBE: Use Support	No	Used to enable / disable support
IBE: Use Support Cart Level	No	Used to enable / disable cart level support

Setting Profile Options for Language and Currency

Set the FND_LANGUAGES and FND_CURRENCY profile options in AOL with the appropriate languages and currencies. Defaults for Oracle iStore 11*i* come from the list of values populated by the values that you entered in AOL for these profile options.

Setting Order Capture (ASO) Profile Options

ASO profile options must be set to enable set up Order Capture to work with Oracle iStore 11*i*. Use the following procedure to set the required profile options.

Steps

1. Log on to Oracle Forms by opening http://<host>:<web port>/, then clicking on **Apps Logon Links > VIS Logon** through the Forms cartridge (UNIX).
2. Log in as SYSADMIN / SYSADMIN.
3. Choose the System Administrator responsibility.
4. Go to **Profile > System > Find System Profile Values**.
5. Check **Application**, **Site**, and **Responsibility**. Enter **Oracle Order Capture** as the application and **IBE_CUSTOMER** as the responsibility.

6. In the Profile field, enter **ASO%**, and click **Find**.
7. Verify and / or set the following Order Capture (ASO) profile options at both the site level and the responsibility level for IBE_CUSTOMER:

Table 7–6 ASO Profile Options

Profile Option Name	Value	Description
ASO: Automatic Numbering	Yes	
ASO: Credit Card Authorization	No	If "Yes," payment is authorized at the time the shopping cart is created. The "No" value is recommended.
ASO: Default Order Type	Standard	
ASO: Default Salesrep	No Sales Credit	
ASO: Product Organization	Vision Operations	
ASO: Quote Order Type	Standard	

Setting Multi Organization (MO) Profile Options

Use the following procedure to set up multi organization profile options.

Steps

1. Log on to Oracle Forms by opening http://<host>:<web port>/, then clicking on **Apps Logon Links > VIS Logon** through the Forms cartridge (UNIX).
2. Log in as SYSADMIN / SYSADMIN.
3. Choose the System Administrator responsibility.
4. Go to **Profile > System > Find System Profile Values**.
5. Check **Site** and **Responsibility**. Enter **IBE_CUSTOMER** as the responsibility.
6. In the Profile field, enter **MO%**, and click **Find**.
7. Verify and / or set the MO: Operating Unit profile option to Vision Operations at the responsibility level for IBE_CUSTOMER.

Setting Up an iMarketing User

Refer to Chapter 5, "Implementing Oracle iMarketing" for instructions on setting up an iMarketing user.

Setting Up Concurrent Manager

Oracle iStore 11*i* includes two concurrent jobs, Store Search Insert and IBE One Click, that are seeded in the ERP concurrent manager request setup.

Store Search Insert

This is a single request concurrent program, that is executed once in the initial setup of iStore, as a post-install step. Run this batch job after you have loaded inventory items in ERP and have set the inventory orgs. This program populates iStore's search table with product information from the inventory tables.

You may need to re-run this program under one of two conditions:

- If you add multiple items that do not appear in the search table.
- If you want to purge all items from the search table.

IBE One Click

This is not a single request job and should be running as a batch process at pre-determined intervals. Schedule this program as a periodically running batch job. This program converts the one-click shopping carts in orders. The programs pick up all the one-click shopping carts, which are not "touched" for the time that is specified in the profile option "One Click Consolidation Time Interval."

Access this setup through ERP forms.

Note: Do not run search insert concurrent program more than once, otherwise it will APPEND records in the search table.

With the concurrent programs, the responsibility "iStore Concurrent Programs" is also seeded. In the initial setup, log in to ERP Forms with System Administrator responsibility, and associate the seeded responsibility "iStore Concurrent Programs" to a user, for example, Vision / Vision98. Another iStore related responsibility is IBE_ADMINISTRATOR, which is also seeded with the Oracle iStore 11*i*. This responsibility is already assigned to user IBE_ADMIN (seeded).

Before running the concurrent program, you need to set profile variables. The concurrent program runs as a different responsibility than the end user responsibility. Establish the following iStore Concurrent Programs Responsibilities, which may or may not match the values of the IBE_CUSTOMER responsibility.

Table 7–7 Profile Option Settings for the Concurrent Program

Profile Option Name	Value
ASO: Default Order Type	Standard
ASO: Default Salesrep	No Sales Credit
ASO: Validate Salesrep	No
IBE: Default Payment Term	IMMEDIATE

Generally, set the same values for the iStore Concurrent Programs responsibility profile options as for the IBE_CUSTOMER responsibility to submit an order.

Running the Concurrent Program to Submit Express Checkout Orders

1. Log into the Oracle Forms application.
2. Select **iStore Concurrent Programs Responsibility** from the list of responsibilities.
3. The "Submit a New Request" window appears (with "Single Request" already selected on the radio button). Click **OK**.

   The "Submit Request" window appears. The first field will be labeled "Name", and there should be a lookup button to the right of that field (three dots). Click it to get a selection of predefined programs.
4. The "Reports" window appears. Select **iStore - One Click Consolidation** and then click **OK**.
5. Set how often and when you want to run this by clicking on the **Schedule** button in the "Submit Request" window.
   - To run right away, select **As Soon as Possible** and then click **OK**.
   - To run once at a scheduled time, click on **Once**, fill in the appropriate fields, then click **OK**.
   - To run regularly, click **Periodically** or **On Specific Days**, fill in the appropriate fields, then click **OK**.

6. Submit the Request by clicking on **Submit**. You are prompted for confirmation and offered to submit another request. Click **NO**.

To Check the Status of the Concurrent Program:

1. Switch responsibility to System Administrator.
2. In the Navigator window, double click on **Concurrent**, and then on **Requests**.
   - The Find Requests window (defaulted to "All My Requests") appears. If the server is not busy, then clicking **Find** may be the fastest way to find your request.
   - If your server is busy, it may be better to enter search criteria and look for "Specific Requests."
3. After clicking **Find**, the “Requests” window displays a list of submitted requests. There should be one (or more) entitled “iStore - One Click Consolidation.” Initially, you may find this in the “green” state with Phase = “pending” or “running.” Click on **Refresh Data** occasionally to check the completion status.
4. Once in the "red" state or phase = "Completed," the "View Output" and "View Log" buttons should become active (if the log and output files have been setup correctly). Use these buttons to find out how many orders the concurrent program was able to successfully submit and how many failed.

Order Management Setup in Forms

Use the following procedure to set up order management in forms.

Steps

1. Switch responsibilities to become the Order Management Super User. (If you don’t have this responsibility, use the System Administrator responsibility to grant it to yourself.)
2. Go to **Setup > Transaction Types > Define**.
3. Search for the Standard transaction type: press **F11**, click in the **Transaction Type** field, enter **S%**, press **Ctrl-F11**.
4. In the Order Workflow field, select **Order Flow - Generic** from the pull-down menu.
5. Click **Assign Line Flows**.
6. Click an empty row in the Assign Workflow Processes region.

7. Enter the following information:

Line Type	UPG_LINE_TYPE_ORDER_1000
Item Type	Standard Item
Process Name	Line Flow - Generic
Start Date	Any valid date (e.g., today's date)

8. Click **OK**.
9. Save the form.

Setting Up Site-Level Profile Options

You must set profile options at the site level.

Steps

1. Switch to the System Administrator responsibility.
2. Go to **Profile > System > Find System Profile Values**.
3. Check **Site** only.
4. Search for the Sequential Numbering profile option.
5. Set the site-level value to **Partially Used** from the LOV.
6. Save the form.
7. Go to **View > Find**, and search for ASO% profiles. Find ASO: Default Salesrep, and select No Sales Credit at the site level from the LOV.
8. Save the form.
9. Switch to the Application Developer responsibility.
10. Go to Profile.
11. Search for IBE%PAY% profiles; find IBE_DEFAULT_PAYMENT_TERM_ID.
12. Make sure that the System Administrator Access is both Visible and Updateable at both the Application and Site levels (i.e., all four checkboxes are checked).
13. Save the form.
14. Switch back to the System Administrator responsibility.

15. Go to **Profile > System > Find System Profile Values**.
16. Search for IBE% profiles. Find IBE: Default Payment Term, and set it to **IMMEDIATE** at the Site level.
17. Save the form.

Order Capture Setup in Forms

Use the following procedure to set up order capture.

Steps

1. Switch to the Order Capture Sales Manager responsibility.
2. Choose **Quote Status Setup**.

   Refer to Order Capture documentation for information on how to set up the Allowed Transition Status region. Create records for each value in the Code LOV (i.e., INACTIVE, LOST, ORDERED, PROBLEM, etc.).
3. Save the form.

Setting Up iStore User Accounts In Forms

Check that the following profile options were set correctly:

IBEGUEST User

1. Go to **Security > User > Define**, and search for the IBEGUEST user.
2. Ensure that the IBEGUEST user has the IBE_CUSTOMER responsibility. If not, add this responsibility and save the form.

See the *Oracle Applications System Administrator's Guide, Release 11i* for information on creating additional users and assigning responsibilities.

IBEADMIN User

1. Go to **Security > User > Define**, and search for the IBE_ADMIN user.
2. Ensure that the IBE_ADMIN user has the IBE_ADMINISTRATOR responsibility. If not, add this responsibility and save the form.

See the *Oracle Applications System Administrator's Guide, Release 11i* for information on creating additional users and assigning responsibilities.

Understanding Cookies

The user session in Oracle iStore 11*i* is controlled and identified by the help of cookies. The cookies are set on the user's browser and are used to identify return customers and other related data. The Oracle iStore 11*i* process is transparent to cookie administration, setup and control. Cookies are managed by JTF foundation methods. If the user turns off browser cookies, JTF makes sure that the information is available through the URL.

The cookie domain is set as the web server domain. For example, "oracle.com" for Oracle's internal store. Once the user registers, the user account is created in the database and is used in the cookies to identify the customer. The Guest user account (Walkin user) is seeded with the product. Personal and Business account types are the basic account types in the store. The business account has the functionality of assigning roles to the business users.