BEA WebLogic Commerce Server Product Family Overview

BEA WebLogic Commerce Server 2.0
Document Edition 2.0
April 2000

Copyright

Restricted Rights Legend

Information in this document is subject to change without notice and does not represent a commitment on the part of BEA Systems, Inc. THE SOFTWARE AND DOCUMENTATION ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. FURTHER, BEA Systems, Inc. DOES NOT WARRANT, GUARANTEE, OR MAKE ANY REPRESENTATIONS REGARDING THE USE, OR THE RESULTS OF THE USE, OF THE SOFTWARE OR WRITTEN MATERIAL IN TERMS OF CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE.

Trademarks or Service Marks

BEA WebLogic Commerce Servers Product Family Overview

Document Edition	Date	Software Version
1.0	April 2000	WebLogic Commerce Server 2.0

Contents

WebLogic Commerce Servers Overview

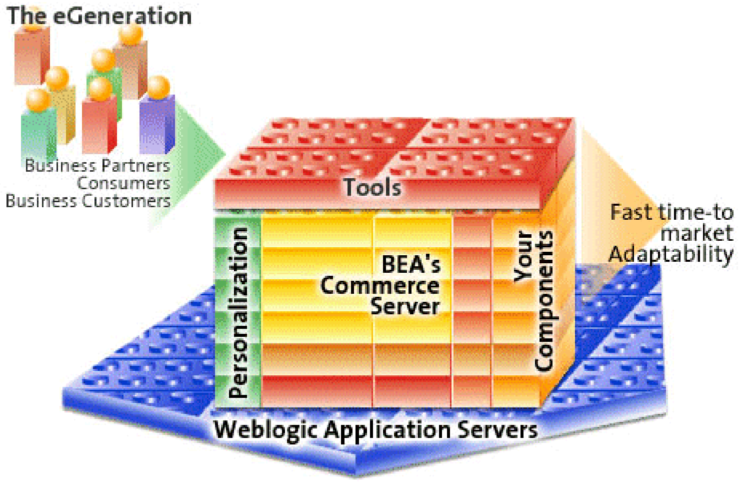
BEA WebLogic Commerce Servers™ (WLCS) enable you to efficiently build, launch, and maintain high-performance e-Commerce sites that offer dynamic, personalized, content delivery.

The Commerce Servers product family includes:

- WebLogic Personalization Server
- WebLogic Commerce Server Components

You can focus on using either product out-of-the-box as a separate toolkit aimed at different levels of e-Commerce design and development, or leverage the full power of the product by incorporating both into your e-Commerce strategy.

The Commerce Servers solution is fully integrated with BEA WebLogic Server™, providing a complete Java-based solution for implementing your e-Commerce site.



WebLogic Personalization Server

The BEA WebLogic Personalization Server (WLPS) is a complete solution for building personalized e-Commerce sites. The Personalization Server is a set of programming code, tools, and utilities for building e-Commerce applications and Web sites that feature customized content delivery based on user profiles and business rules.

Do you want to learn more about the Personalization Server? Please read on . . .

- What is personalization?
- What are the key components of the Personalization Server?
- Personalization Advisor
- Content Retrieval and Content Management
- Rules and Rules Management
- JSP Tag Library
- Portals and Portlets
- Properties and Property Sets
- User and Group Management
- How do I use the Personalization Server to create an e-Commerce site that delivers personalized content?
- Ready to get started with the Personalization Server?

What is personalization?

Personalization is the means by which Web content developers can tailor an application to a particular individual or group based on any number of criteria. The criteria can be predefined user attributes such as age and gender, or can be based on behavioral information gathered as the user navigates a site.

Using the Personalization Server, you can build Java-based Internet pages and sites with dynamic, personalized document content. You can customize what content gets delivered based on individual user profiles. The Personalization Server has a built-in rules editor that you use with Java Server Pages (JSP) tags to deliver a responsive, customized experience for users.

What are the key components of the Personalization Server?

The key components of the Personalization Server are:

- Personalization Advisor
- Content Retrieval and Content Management
- Rules and Rules Management
- JSP Tag Library
- Portals and Portlets
- Properties and Property Sets
- User and Group Management

Personalization Advisor

In the BEA WebLogic Personalization Server (WLPS), the Personalization Advisor provides content personalization capabilities by using an embedded rules engine to classify a user and create a dynamic query into a content database to return *personalized* content for that user. The Personalization Advisor is a stateless session bean that gives personalized advice and recommends content. It can retrieve any type of content in a document management system and display it with a JSP page or servlet. The session bean can be accessed directly, but is typically accessed through a set of Java Server Page (JSP) tags. These JSP tags enable HTML developers to assemble dynamic pages without writing any Java code.

The Personalization Advisor ties together the core Personalization Server services in WLPS. These services are user profile management, rules service, content management, and the WLPS foundation. The Personalization Advisor gathers information from the user profile provided by the user management component, submits that information to the rules service, runs the resulting queries against the document management system used in the content management component, and returns the content to the JSP developer.

The Personalization Advisor component includes a JSP tag library and a personalization advisor EJB stateless session bean that accesses the core personalization services. JSP developers can take advantage of the personalization engine using the JSP tags, and EJB developers can access the underlying session EJB and its features via the public Personalization Advisor bean interface.

The Personalization Advisor is designed to *scale* for large e-Commerce web sites. It uses a pooling and caching mechanism to keep rules engines ready for rapid evaluation of Personalization rules.

For more about the Personalization Advisor, see the topic Creating Personalized Applications with the Personalization Advisor.

Content Retrieval and Content Management

The Personalization Server has a content management feature that provides access to content through both tags and EJBs. The content management tags enable a JSP developer to receive an enumeration of content objects by querying the content database directly with a search expression syntax. The content management tools are designed to be used with the rules manager and rules services, user profiles, and property set manager to create personalized content delivery on an e-Commerce site. The primary interface to the functionality of the content management component is the ContentManager session bean.

The Content Advisor is a key Personalization Server component that allows a J2EE Web developer to personalize content for users of a Web site or a personalized application. The Content Advisor includes JSP tags, tag library, and a Content Advisor EJB stateless session bean that accesses the core personalization services: Foundation, User profile management, rules service, and content management The tag library and session bean contains personalization logic to access these services, sequences personalization actions, and returns personalized content to the Web site. This architecture makes the details of the Personalization engine optionally transparent to the JSP developer; only knowledge of the Content Advisor JSP tags is required. For more advanced JSP development, access to the underlying Content Advisor EJB and all of its features is available. JSP and Java developers can use the Content Advisor to deliver any type of personalized content either in a portal or simply as standard HTML output.

What is content management?

Content management is the process of creating, classifying and then incorporating content into your application. This often involves a workflow process for coordinating the creation and classification process. Content management also requires a search capability. Typically, content is defined as documents and images. Our definition of content is any data that can be matched to a user, searched for, and presented. This includes everything from banner ads, informational documents, and products in a catalog to investments instruments. To accomplish this we have abstracted out the interfaces to content in a separate package and created specialized versions of those interfaces for document management.

The WebLogic Personalization Server provides content retrieval and supports third party content management solutions:

- Interwoven Teamsite/OpenDeploy
- Documentum 4i
- Macromedia Dreamweaver

Content management tools such as Documentum 4i and Interwoven Teamsite/OpenDeploy provide functionality related to document management and the workflow of creating and classifying content.

Macromedia Dreamweaver provides HTML authoring capabilities. (Note that Dreamweaver JSP extensions will be available shortly after the initial WebLogic Commerce Servers 2.0 April release.)

For more about using content management in the WebLogic Personalization Server, see the topic Creating and Managing Content.

How do I choose what content management tools to use?

- **WLPS Bulk Loader**. For sites with limited content personalization needs and existing meta-tagged HTML, WLPS includes a bulk loader. The bulk loader can parse a directory of HTML files and store their URL address and meta-data attributes in a JDBC store. The bulk loader automatically creates the schema for these attributes.

- **Interwoven Teamsite/OpenDeploy Integration**. For customers who have larger amounts of content and want more control over the publishing and tagging of content, WLPS provides integration with the Interwoven Teamsite/OpenDeploy product. Teamsite is a content capture, versioning, staging, and publishing system. Teamsite is not a run-time engine; it does not actually get queried by the Personalization Server. Teamsite works with the Personalization Server to publish content to a well-known location.

  A Personalization Server customer can purchase the Interwoven Teamsite 4.0 product for content management and publishing. Teamsite templates define the meta-data attributes for the content; customers can use the templates to categorize the documents during the creation process. Once captured, a Teamsite administrator can use the Interwoven OpenDeploy workflow capability to publish the documents to a known file system and the meta-data to our JDBC store.

  For information on using Interwoven Teamsite/OpenDeploy with the WebLogic Personalization Server, see the topic Creating and Managing Content. The extensions for Interwoven are available on the BEA download center as one of the Commerce Servers download options.

- **Documentum 4i Integration**. As a compatible document management solution, we provide integration with the Documentum 4i product. Documentum products manage the meta data and documents in their own repositories.

  Personalization Server customers can purchase Documentum 4i to manage the documents or content for their e-Commerce site. Content developers working with Documentum create meta-data capture tools in Documentum 4i. When they complete the authoring process and checkin their documents to Documentum, they must tag the content. Depending on administrative options, this content can be available immediately to the Personalization Server for display on the customer’s e-Commerce site.

  For information on using Documentum 4i with the WebLogic Personalization Server, see the topic Creating and Managing Content. The JDBC driver for Documentum 4i is available on the BEA download center as one of the Commerce Servers download options.

- **Macromedia Dreamweaver Integration**. Shortly after the April 2000 WebLogic Commerce Servers 2.0 release, JSP extensions for MacroMedia

Dreamweaver will be available on the BEA download center as one of the Commerce Servers download options. Please check back soon at the BEA Download Center to get the JSP extensions. This integration will allow drag and drop of Personalization Server JSP tags using a Dreamweaver pallet.

Rules and Rules Management

Rules and Rules Management form an integral part of the personalization process. The business logic encompassed by rules and managed through the Rules Administration tool enable the delivery of personalized content marketed specifically to to fit individual user profiles. The Rules Management tool provides editing, deploying, and run-time services for delivering personalized content to end-users as they browse a Web site. These personalization components use business rules to match users and groups with appropriate content. The rules components are what provide the glue between the user and the content that is to be provided.

This Rules Management component includes two major parts: a rules editing tool with a graphical user interface that allows developers to define classification and content selection rules, and a run-time service that matches users with content based on these rules.

The rules editor is an HTML based editor for creating a structured system of business rules. The rules are used along with user profiles to personalize the content delivery of an e-Commerce site. The syntax of the rules is rules engine-independent. The output of the rules editor is a *RuleSheet*, stored in an XML format.

For more about Rules and Rules Management in the WebLogic Personalization Server, see the topic Creating and Managing Rules.

JSP Tag Library

Java Server Pages are a J2EE component for generating dynamic Web page content. The Personalization Server features an extensible JSP tag library that you can use to develop portlets, rules, user profiles, and properties.

For a reference of JSP tags, see the JSP tags reference.

Portals and Portlets

The Portal provides an HTML windowing toolkit for web site developers, group administrators and actual end-users to "personalize," customize and individualize the layout, look, and content of an e-Commerce site. It provides these features through a set of JSP tags, EJBs and administration tools. If the customer chooses to use the Portal product, they will develop "portlets" in JSP. These "portlets" are mini-windows on to information, content or application services available in the portal. They can be minimized, maximized in their own window, edited, and provided with help. Once they are built, a Portal administrator selects "portlets" for availability in their portal. Once available, a portal user controls the layout and visibility of these windows. The portal provides a framework for one-, two-, and three-column layouts.

For more about using Portals and Portlets in the WebLogic Personalization Server, see the topic Creating and Managing Portals.

Properties and Property Sets

The Personalization Server provides the notion of a property set, which is a construct to help group properties that are related to each other. Properties are named attributes that may have developer-defined default values. For example, a health-related property set might consist of the properties age (integer), height (float), weight (float), diabetic (boolean). Property sets are often associated with objects that extend ConfigurableEntity, to describe the valid range of properties for that object.

The Property Set Management tool provides schema details to Personalization Server subsystems such as User Management and Rules Management. Both group and user schemas can be created in the Property Set Management tool. Such profile schemas prescribe sets of attributes required by user profiles.

Many significant objects within the internals of the Personalization Server extend an EJB entity object type called ConfigurableEntity, whose main purpose is to hold values for a set of properties that may change over time. A ConfigurableEntity provides facilities for defining properties in a flexible way, so the range of properties can be defined added to, deleted from, and modified over the course of time. Each property is defined as one of a predefined set of native personalization types, such as boolean, text, and float.

For more about using Portals and Portlets in the WebLogic Personalization Server, see the topic Creating and Managing Property Sets.

User and Group Management

User Management combines enterprise data about a user with profile data used to personalize the user's view of the application. This process includes mapping the pre-existing data and defining a schema for the personalization attributes. User and Group Management is provided through a set of tools in our administration application and alternatively through any WebLogic Server Realm.

Typically, new e-Commerce sites that rely on self-registration will rely on the WLPS user management tools. For sites with large collections of existing uses in the form of customers or employees, the WebLogic Realm support can provide added convenience and security through products like LDAP 3.0 servers. Information such as the customers' billing address and accounts status, and an employee's date of hire and benefits selections are examples of data that reside in a corporate database or a Lightweight Directory Access Protocol (LDAP) server. In most cases this information is well known and modeled as an EJB.

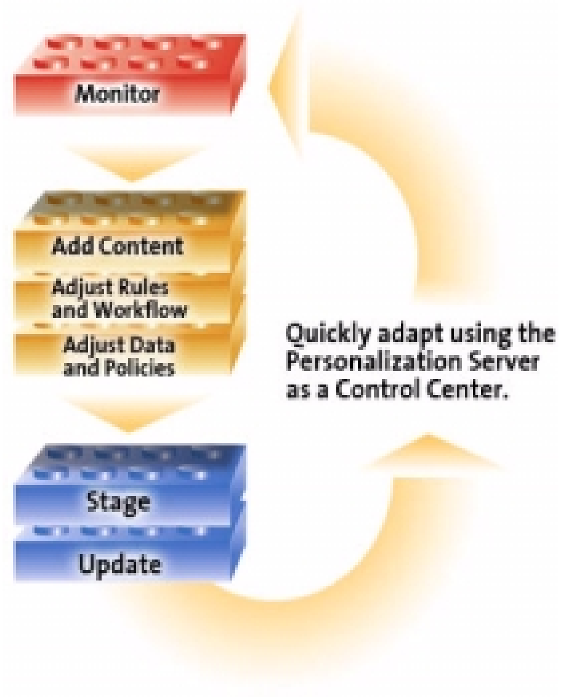
For more about using user and group management in the WebLogic Personalization Server, see the topic Creating and Managing Users.

How do I use the Personalization Server to create an e-Commerce site that delivers personalized content?

Using the Personalization Server tools to create and deploy new e-Commerce sites involves different tasks and phases. In general, during each stage of the e-Commerce development cycle, developers and system administrators complete tasks to create and deploy WLCS 2.0 to deliver personalized content to end-users. The following table provides a quick look at the activities and roles that are needed to build your personalized e-Commerce presence on the Internet.

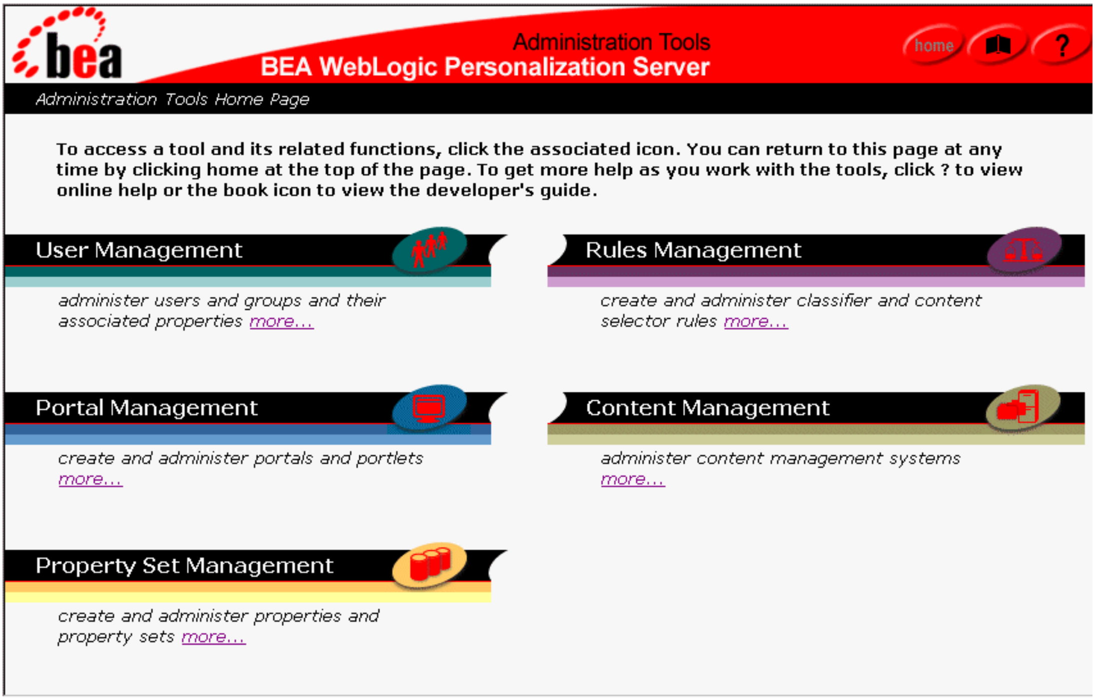
Stage	Roles	Phases	Tasks
Develop	♦ Business analysts ♦ JSP developers ♦ Java developers ♦ HTML content authors	1. Analyze personalization needs. 2. Design user profiles and content. 3. Build profiles, content, and rules.	♦ Develop user profile schemas. ♦ Create and organize content. ♦ Write rules to match user profiles and content.
Deploy	♦ System administrator Web manager	4. Deploy commerce site. 5. Maintain and administer users and groups.	♦ Stage the commerce site ♦ Monitor site usage and user profiles
Adapt	♦ System administrator ♦ Web master	6. Monitor site usage. 7. Adapt profiles and rules to changing business strategies.	♦ Update content ♦ Revise and update rule definitions ♦ Changing workflows.

As shown in the following illustration, the Personalization Server is an ideal solution for developing and maintaining your e-Commerce site in terms of the *adaptive life cycle* process (develop, deploy, adapt), which is required to create a truly up-to-date and responsive site in today’s Web market place.


Monitor
Add Content
Adjust Rules and Workflow
Adjust Data and Policies
Stage
Update
Quickly adapt using the Personalization Server as a Control Center.

How do I work with the Personalization Server tools?

The Personalization Server tools enable you to create an e-Commerce site in an integrated and readily accomplished development activity. For most of your tasks you will be working with the BEA Personalization browser-based GUI. You will also be working with JSP tags and some content management tools. The following illustration shows the some the the primary browser tools you will work with to incorporate personalization into your e-Commerce site.



Administration Tools
BEA WebLogic Personalization Server

Administration Tools Home Page

To access a tool and its related functions, click the associated icon. You can return to this page at any time by clicking home at the top of the page. To get more help as you work with the tools, click ? to view online help or the book icon to view the developer's guide.

User Management

administer users and groups and their associated properties more...

Rules Management

create and administer classifier and content selector rules more...

Portal Management

create and administer portals and portlets more...

Content Management

administer content management systems more...

Property Set Management

create and administer properties and property sets more...

The following example illustrates the high level process you might go through to develop your personalized e-Commerce site.

How CompanyZ Launches an e-Commerce Site

1. By working with the **User Management tool**, a personalization administrator would create user profile schemas.

2. If no **content management** system such as Interwoven or Documentum is used, the content can be written in HTML. Content authors would embed meta tags within the HTML. These meta tags is associated with content bound for personalization. A bulk loader would search the HTML for meta tags. Once found the meta tags would be indexed within metadata database.

3. Using the **Rules Management tool**, personalization administrators create rules that associate the user profiles to content metadata.

4. Once the appropriate metadata is matched with a user, the content associated with the metadata is sent to **Portal** to be viewed by the user.

Features at a Glance

- Personalization editor
- Personalization advisor
- JSP tags for personalization, content, and content management
- Personalized portlets
- Extendible user profiles
- Realm-based user management
- Embedded rules engine
- Three content management options
- Dreamweaver Drag and Drop tags

Ready to get started with the Personalization Server?

The Personalization Server is a complete solution that enables rapid deployment of adaptable and personalized e-Commerce applications, allowing businesses to extend competitive advantage and accelerate response time to customer and market demands.

With the Personalization Server you can build a wide range of portal types, from business-to-consumer “megaportals” to business-to-business enterprise portals. What this means for your e-Commerce enterprise is a flexible, personalized, Web presence that listens and responds to your customers and partners based on what you define as important factors.

Now that you have an understanding of all the elements that WLPS provides to give you personalization, you should dive right in to develop a sample e-Commerce site. A good way to get started is to take a tour through our MyBuyBeans personalization tour. We actually provide you with two samples that you can use:

- Take the MyBuyBeans Personalization Tour, to learn how to use the robust rules and content management features of the Personalization Server
- Work with the Acme Portal Example, to learn how to build a portal.

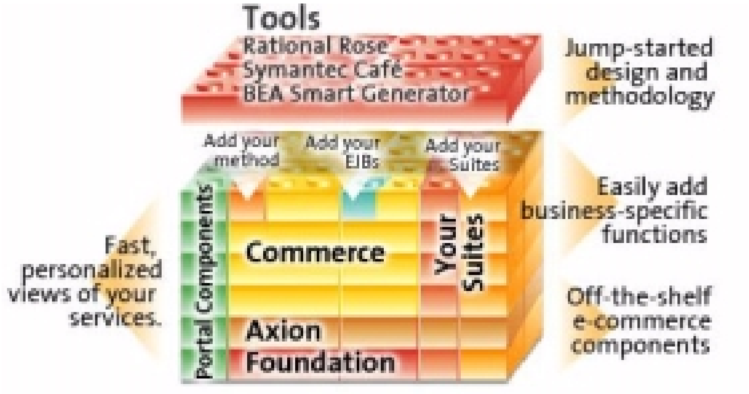
WebLogic Commerce Server Components

The BEA WebLogic Commerce Server (WLCS) components allow companies to quickly assemble extensible and scalable e-Commerce solutions. The kit includes more than 80 pre-built, pre-tested components such as customer, session, order, item, shopping advisor, trouble ticket, inventory, and invoice.

Together with a robust set of foundation classes, the WebLogic Commerce Server provides up to 80% of common e-business functions and allows enterprises to focus on building their unique competitive advantage.

Do you want to hear more about components? Please read on . . .

- Build versus Buy
- What are Commerce Server components?
- How does the Commerce Server object model and the Smart Generator speed EJB development?
- Industry-Standard Design Patterns are built in to components.
- Features at a Glance
- Specifications
- Ready to get started with Commerce Server components?


Tools
Rational Rose
Symantec Café
BEA Smart Generator
Jump-started design and methodology
Add your method
Add your EJBs
Add your Suites
Easily add business-specific functions
Portal Components
Commerce
Your Suites
Axion
Foundation
Fast, personalized views of your services.
Off-the-shelf e-commerce components

Build versus Buy

In the process of planning e-Business applications, corporations are faced with a build or buy decision. Building an application of this kind from the ground up would consume considerable time and resources. On the other hand, an off-the-shelf application does not meet the company's unique needs. The best solution is to use *components*. Components are packages of pre-built business functions that jump-start the development of an e-Business application. Components allow developers to customize and snap together enterprise applications quickly, while tailoring them to specific business needs. The result is a complete solution that takes advantage of EJB technology without its time-consuming complexities.

What are Commerce Server components?

Commerce Server components are software building blocks for e-Business that can be selected and snapped together to create a robust e-Commerce Web presence. With WebLogic Commerce Servers components developers do not need to start from scratch or master EJB complexities. By using WLCS components, they can build e-Business applications customized for a company's unique business needs in record time frames.Components can be used as is, or customized or extended it to fit particularly unique aspects of your business scenario.

The components themselves are Enterprise JavaBeans (EJB) built to the EJB 1.1 specification for a scalable, robust, industry-standard solution. The WLCS components family is structured as packages, each of which contains a set of components that plays a specific role in assembling an enterprise application. WLCS provides *industry specific component packages* for financial services, telecommunications, and Internet retail. Today WLCS includes component packages required to build **e-business** applications, with an emphasis on Internet e-Commerce and customer self-service.

For a comprehensive look at the components offered and the features they provide, see the full-fledged WebLogic Commerce Server components overview and the list of all component packages in the Components Catalog.

How does the Commerce Server object model and the Smart Generator speed EJB development?

Java developers can quickly create coherent, well-designed EJB components by modeling them based on our UML model and then using our Smart Generator to generate Java source code for the EJBs.

Code generation from UML has long been recognized as a promising technology. This technique is powerful because it allows the designer to model the components in a natural way without being concerned with implementation-specific details.

The Commerce Server development tools go the extra mile with this technique to provide an easy way to generate the implementation of business logic, provide for iterative development, and maintain a synchronization between the code and the model. To this end, the development utilities do not assume a direct mapping from the model to the underlying language constructs. The user models the business objects and the Smart Generator creates a set of classes that implements these objects with reference to the Enterprise JavaBeans 1.1 Specification. Many of the laborious tasks of creating access methods and handling containment of references is automatically handled.

Industry-Standard Design Patterns are built in to components.

WebLogic Commerce Servers components interact with each other using industry standard Design Patterns to solve a wide range of business problems. The use of proven design patterns such as Strategy, Policy, Chain of Responsibility, and many more results in a better business model, lowered cost of development and maintenance, and faster time-to-market.

For more about the design patterns built in to our components, see the WebLogic Commerce Server components overview.

Features at a Glance

- Customizable Enterprise JavaBeans built from the ground up
- Plug-and-play components that allow you either to use our out-of-the-box solution, or to integrate with your legacy applications
- Easy-to-use component APIs that are fully customizable and extensible using technologies like pluggable methods and dynamic runtime configuration
- Implemented using established design and analysis patterns for ease of use and re-use
- WLCS architecture ensures that applications built on the components model run in a scaleable, high-performance, enterprise-class fashion
- Components work with other EJBs, including third-party and custom-built components
- High-performance features such as pass-by-value (PBV)
- Works with leading EJB Application Servers
- Works with leading databases (Oracle, Sybase, DB2, Cloudscape, etc.)
- Integrate with legacy systems (CICS, IMS, legacy databases)
- You don’t need to be an EJB expert to use and customize our pre-built EJBs!

Specifications

- 100% Pure Java
- Components are Enterprise JavaBeans 1.1
- Support for Java 2
- Support for the advanced Java 2 Collections API
- Support for Enterprise Java APIs including EJB, RMI, JNDI, JTS, and JDBC
- Support for modeling using UML, with roundtrip engineering from Rational Rose
- Component can be invoked from Java Clients, Java Servlets, Java Server Pages (JSP/JHTML), CORBA Clients and Servers
- Support for ActiveX/COM, and other clients (Visual Basic, PowerBuilder)
- Full support for BEA WebLogic Server and related features, including clustering and JDBC connection pooling. Support for other leading application servers coming soon.

Ready to get started with Commerce Server components?

If you would like to see a working example of how to snap together the WebLogic Commerce Servers components to form a robust, high-performance, retail Web site, see the WebLogic Commerce Servers Components MyBuyBeans Tour. The tour steps you through the development and deployment process using the imaginary MyBuyBeans.com retailer as a focus.

For a detailed look at the WebLogic Commerce Server EJB components development process, and the use of the Smart Generator and UML models, see the Component Development Process.

For information about deployment and bean-managed persistence, see Deploying Your Application.