



IBM Software Group

WebSphere® Commerce V6

Updating WebSphere Commerce stores



@business on demand.

© 2007 IBM Corporation
Updated September 18, 2007

Welcome to the WebSphere Commerce V6 presentation. This presentation describes the administrator's role in publishing updates to WebSphere Commerce stores.

Unit objectives

- Use the loader utilities to upload data
- Describe the use of staging utilities
- Describe the use of authoring servers and workspaces
- Deploy simple store customizations and updates using WebSphere Application Server V6

This presentation discusses some of the WebSphere Commerce utilities that you have at your disposal to aid in creating and updating your site.

Store data overview



WebSphere Commerce store data is classified into five categories, depending on how it is created and managed.

When a WebSphere Commerce server instance is created, the database is populated with the required instance data. This is also known as the bootstrap data. This data mostly includes site default settings that do not change or can be overridden at the store level.

When a store archive is published, the core data, configuration data, and managed data are created. Core data and configuration data do not typically change unless functionality of the site changes. The seller may change managed data as store policies and offerings change over time. A content management system is required to update managed data.

Operational data is created dynamically as customers interact with the store. This data may need periodic cleanup to remove obsolete data.

Tools for creating store data

- WebSphere commerce browser based tools
 - ▶ Administration console
 - Configure messaging, scheduler, and so forth
 - Publish store archive
 - ▶ Organization administration console
 - Manage organizations, users, member groups, roles, and access policies
 - ▶ WebSphere commerce accelerator
 - Edit store profile, tax, shipping, and fulfillment centers
 - Manage catalog, marketing campaigns, promotions, auctions, inventory, fulfillment, accounts and contracts, customers, orders, payments
 - ▶ Workspace administration tool
 - Manage workspaces and tasks
- Mass loader tools
 - ▶ Administrative tools - transform data
 - ▶ Loader package - prepare and batch load data
- Integration
 - ▶ Use inbound messages to run commands to manage store data

WebSphere Commerce includes a number of tools, which can create and manage store data. Most of these tools work with the managed data. While the browser-based tools can be very useful for adding and managing data, you would probably not want to use them to add large amounts of data.

Creating store data using XML

- If you want to use a store archive:
 - ▶ Examine the XML files in a starter store archive and associated DTDs in `<WC_HOME>/wc.ear/xml/sar` directory
 - ▶ Add the data to the existing XML files
- If the data does not already exist:
 - ▶ Create data as XML files using the WebSphere Commerce XML schema
 - ▶ See DTD `<WC_HOME>/schema/xml/wcs.dtd`
- If the data already exists in some format:
 - ▶ Export data to XML or delimited ASCII files
 - ▶ Use tools to transform the data to the WebSphere Commerce XML schema

The XML files can then be loaded into the database using the loader package.



Most stores are likely to have existing applications and tools, which will provide some of the data that WebSphere Commerce needs. This is most common for catalog data, but other data such as users, tax and shipping calculations, or fulfillment centers can also exist in the systems. WebSphere Commerce supports loading of data in XML format using the WebSphere Commerce loader package.

If you will use a custom store archive as a means of creating store data, you should examine the XML files in one of the starter store archives and their corresponding DTD files to understand the XML format that the data needs to be in. Your data can be added to the existing XML files in the store archive, and then you can publish the store archive to update the database.

If you do not want to use the store archive but you will be creating data from scratch and want to load it into the database, you should create XML files that conform to the WebSphere Commerce XML schema. The schema is defined in the `wcs.dtd` file that can be found in `<WC_HOME>/schema/xml` directory. You need to create XML files, which conform to this DTD and use the loader package to load the data.

If you already have data in an application such as a spreadsheet, you need to export the data and transform it into a format that WebSphere Commerce can use. Most applications support export as delimited ASCII text and some applications support XML export. The loader tools can be used to transform the data to the WebSphere Commerce XML schema and load it into the database.

Loader package overview

- Text transformer
 - ▶ Transforms delimited text to XML format
- XML transformer
 - ▶ Transforms an XML file from one schema to another
- DTD generator
 - ▶ Creates a DTD from a list of database tables
- ID Resolver
 - ▶ Substitutes symbolic identifiers in an XML document with numeric identifiers determined from the database
- Mass loader
 - ▶ Loads data from a resolved XML file
- Extractor
 - ▶ Extracts data from a database to an XML file
- File loader
 - ▶ Loads managed files into the database

The loader package provides various tools for transforming and loading data into the WebSphere Commerce database.

The **text transformer** and **XML transformer** can be used to transform data from existing applications so that it can be imported into the WebSphere Commerce database.

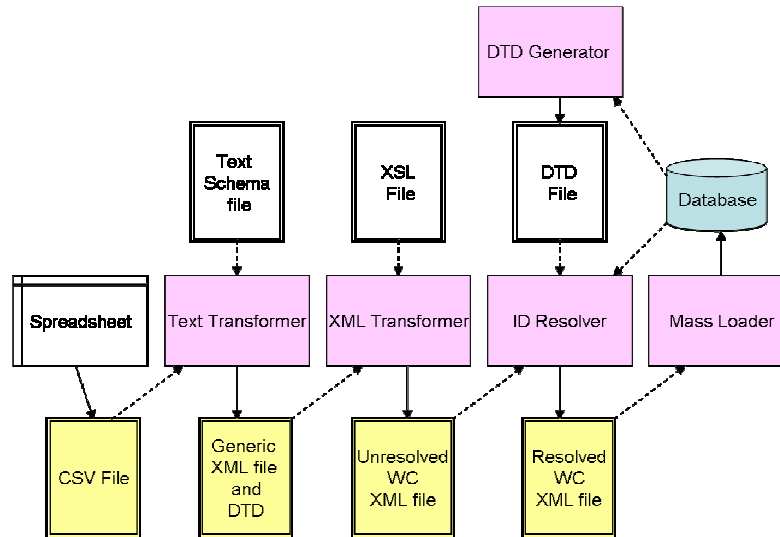
The **DTD generator** creates an XML Document Type Definition (DTD) for a list of database tables by querying the database table structure and constraints.

Normally, when you create an XML file for loading into the database, you should not add the numeric identifiers for the primary key columns in the database tables. This makes the XML file easier to repurpose with other databases, which may have different identifiers for the same data. The **ID resolver** can determine the appropriate numeric identifiers to use for existing or new data and create a resolved XML file for the loader.

The **mass loader** loads data from a resolved XML file into the database.

The **extractor** extracts data from the database into an XML file. Then the XML transformer or Text transformer can be used to transform it into a format that can be imported into an external application. This makes it easier to exchange data in both directions between applications and WebSphere Commerce.

Transform, resolve, and load



7

Updating WebSphere Commerce stores

© 2007 IBM Corporation

Depending on how the data you want to load into the WebSphere Commerce database has been created, you may need to combine multiple loader packager tools to transform and load the data.

Many applications, such as spreadsheets, support output to a comma separated variables text file. In this case, you would need to create a text schema file and run the text transformer to transform the data to generic XML. Then use an XSL editor to create an XSL file and run the XML transformer to transform the generic XML to an unresolved WebSphere Commerce XML file. Then run the ID resolver to resolve the symbolic identifiers to their numeric values and run the mass loader to load the data into the database.

If your application supports XML output but it uses a different schema than WebSphere Commerce, you would begin with the XSL editor and XML transformer.

If you are creating unresolved XML data directly from the WebSphere Commerce schema, you could start your process with the ID resolver.

The wcs.dtd file contains definitions for all of the WebSphere Commerce database tables, but if you need to support custom tables, you may need to run the DTD generator.

Running the loader package tools

Batch files are provided in the <WC_HOME>/bin directory

| | |
|------------------|------------------|
| Text Transformer | txttransform.cmd |
| XML Transformer | xmltransform.cmd |
| ID Resolver | idresgen.cmd |
| DTD Generator | dtdgen.cmd |
| Mass Loader | massload.cmd |
| Mass Extractor | massextract.cmd |
| File Loader | fileloader.cmd |

This slide lists available loader package tools. When you run the loader package tools, you must specify the appropriate command line parameters. WebSphere Commerce Information Center contains documentation for the command syntax.

Resolving identifiers

- XML entities can be used for common identifiers (use ɾ)
 - ▶ Example: member_id="&MEMBER_ID;"
 - ▶ Replacement value can be defined in a DTD
- ID resolver symbols determine unique identifiers (use @)
 - ▶ Example: catalog_id="@catalog_id_1"
 - ▶ Replacement values are determined from primary keys or unique indexes in the database.
 - ▶ If no unique index exists on a table, additional hints can be provided in IDResolveKeys.properties

9

Updating WebSphere Commerce stores

© 2007 IBM Corporation

The XML file containing the data to be imported can use XML entities and ID resolver specific symbols instead of hard coding ID numbers. This makes it easier to reuse the file without worrying about conflicting identifiers.

XML entities begin with ampersand sign (&) and end with semicolon (;). ID resolver symbols begin with "at" sign (@). ID resolver symbols are placed in a field, which corresponds to the primary key or unique index within a database table, and wherever the symbol appears, the appropriate identifier will be substituted.

Most tables have a unique index so it is easy to resolve IDs. Some tables, for example INVENTORY, have no unique index. A Java properties file, IDResolveKeys.properties, can be used to specify additional information for the ID @if, which will help it to resolve keys for tables without a unique index.

Managed files

Managed file: *Any file uploaded to WebSphere Commerce using the catalog and marketing attachments tools or the fileloader utility*

- Written to the database first and then copied to the WebSphere Commerce EAR.
- EAR updater is called by the scheduler.
- Use the Store Preview function to view the file in your store before it has been propagated to the EAR.

Any file uploaded to WebSphere Commerce using the catalog and marketing attachments tools or the fileloader utility is called a managed file. The catalog and attachments tools can be found in WebSphere Commerce Accelerator.

Certain parameters must be met before the managed file is copied to the EAR. The default behavior is to wait until at least 10 files are waiting or one hour has passed.

In a federated environment, all copies of the WebSphere Commerce EAR file are updated automatically. Updating all WebSphere Commerce nodes in a federated environment is not an instantaneous process. The exact amount of time taken to complete the update depends on the size of the managed file and the size of WebSphere Commerce EAR file.

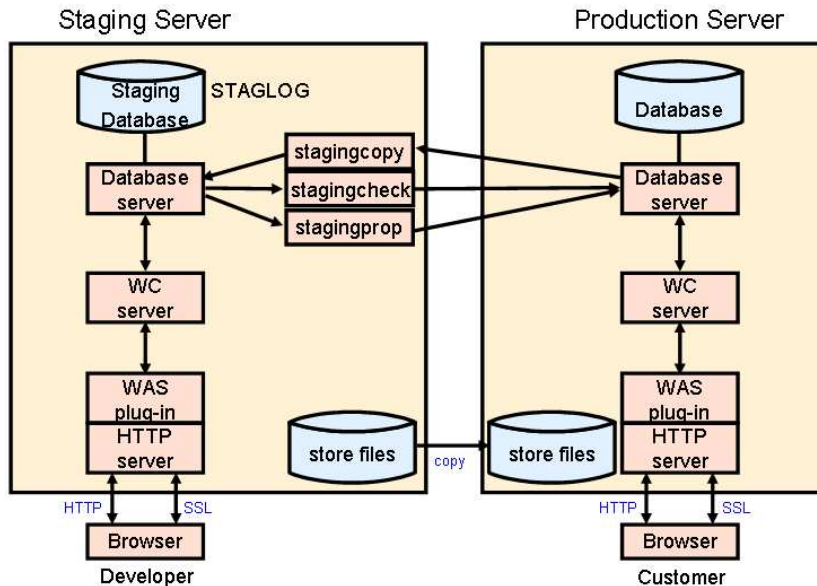
Staged deployment

- Developer will create/customize store assets
 - ▶ Storefront assets (JSPs, HTML, images)
 - ▶ Business logic (EJB/command Java code)
 - ▶ Store data assets
- Staging server allows system changes to be tested in a safe environment
 - ▶ Copy production data to staging database
 - ▶ Update staging server and test changes
 - ▶ Check for unique index conflicts with production database
 - ▶ Propagate changes back to production database
- File assets need to be copied manually
 - ▶ Refer to programmer 's guide for deployment process

Most e-commerce sites operate 24 hours a day, 365 days of the year. This schedule makes it difficult to perform maintenance or to test changes to the system.

The WebSphere Commerce Staging Server facility allows a merchant to make a copy of their live production server database so that they can test updates without affecting customers. This function is useful for testing updates to the product catalog, and for testing new shopping process commands and pages. The **fileprop** utility can be used to copy managed files from staging environment to production environment.

Staging server



12

Updating WebSphere Commerce stores

© 2007 IBM Corporation

A staging server is a dedicated test machine which allows site and store functionality to be tested without needing to bring down the production server and without the risk of untested data being made available on the production server. The staging server is a second WebSphere Commerce instance that typically runs on a dedicated development machine in the merchant's intranet.

The staging server consists of components divided into four categories.

WebSphere Commerce Server application server environment is a standard instance with your store data and files.

Delta data capturing function consists of extra tables and database triggers that log changes to certain tables.

Staging server data includes configuration and operational data. Operational data (like customer information and orders) will not be copied to the production server.

Staging server utilities move data between staging and production environments.

Staging tables

- Tables supported by staging are grouped into three scopes:
 - ▶ Site data scope
 - Includes access control, country, state, approval flow, language, quantity, currency, role, and various data types
 - ▶ Merchant data scope
 - Includes access control, command, store, catalog, inventory, calculation, contract, policy, currency, tax, shipping, returns, and vendor data
 - ▶ Site and Merchant data scope
 - Includes command, member, fulfillment center, and store group data
- Customer and other operational data is not supported
- Support for custom tables can be added

The database tables managed by a staging server include those that contain the operational data of the site and stores but do not include any customer, order, payment, or instance related data. This allows the merchant to set up dummy users and make dummy orders for testing their store rules without affecting real customers.

To configure the staging server for custom database tables, perform the following steps.

First, identify your custom table scope (site data, merchant data, or both).

Second, create the triggers for your database table using the corresponding trigger examples based on your table's scope and index type.

Third, insert the custom tables into the STGSITETAB, STGMERTAB, and STGMRSTTAB database tables based on the scope of the data.

Staging utilities

- Stagingcopy
 - ▶ Copies data from the production database to the production-ready data on a staging or authoring server
- Stagingprop
 - ▶ Copies data and managed files from the production-ready data on a staging or authoring server to the production database
- Fileprop
 - ▶ Publishes managed files from the production ready data to the WebSphere Commerce EAR
- Stagingcheck
 - ▶ Checks for unique index key conflicts between production-ready data and the data on the production server

Note: You cannot use these commands to create or update Request for Quote objects.



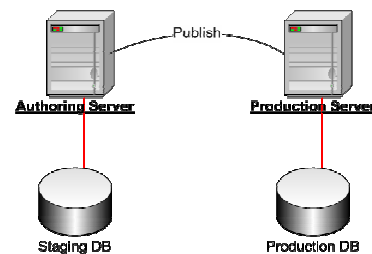
Staging copy is normally only run once to create the initial base of data for the staging server. From that point on, all changes should be made to the staging server first and then propagated to the production server. You cannot use the stagingcopy utility at all if request for quote objects exists on your production system.

Staging check is normally run before propagating data from the staging or authoring server to the production server. If there are no conflicts, the staging propagate utility will run. If the appropriate command line parameters are provided, staging propagate will also copy managed files after copying data.

If there is no data to be propagated, the fileprop utility can be used to publish managed files.

Authoring Server

- **Authoring server:**
 - ▶ Consists of a staging server with workspaces enabled
 - ▶ Allows business users to control site changes in the areas of product, marketing, and merchandising management
- **Workspaces:**
 - ▶ Isolates access-controlled work areas
 - Separate from production-ready data for the same WebSphere Commerce instance
 - ▶ Divides work into task groups and tasks
 - Task group is smallest set of data to commit to production-ready data
 - Task is used to assign work to users



15

Updating WebSphere Commerce stores

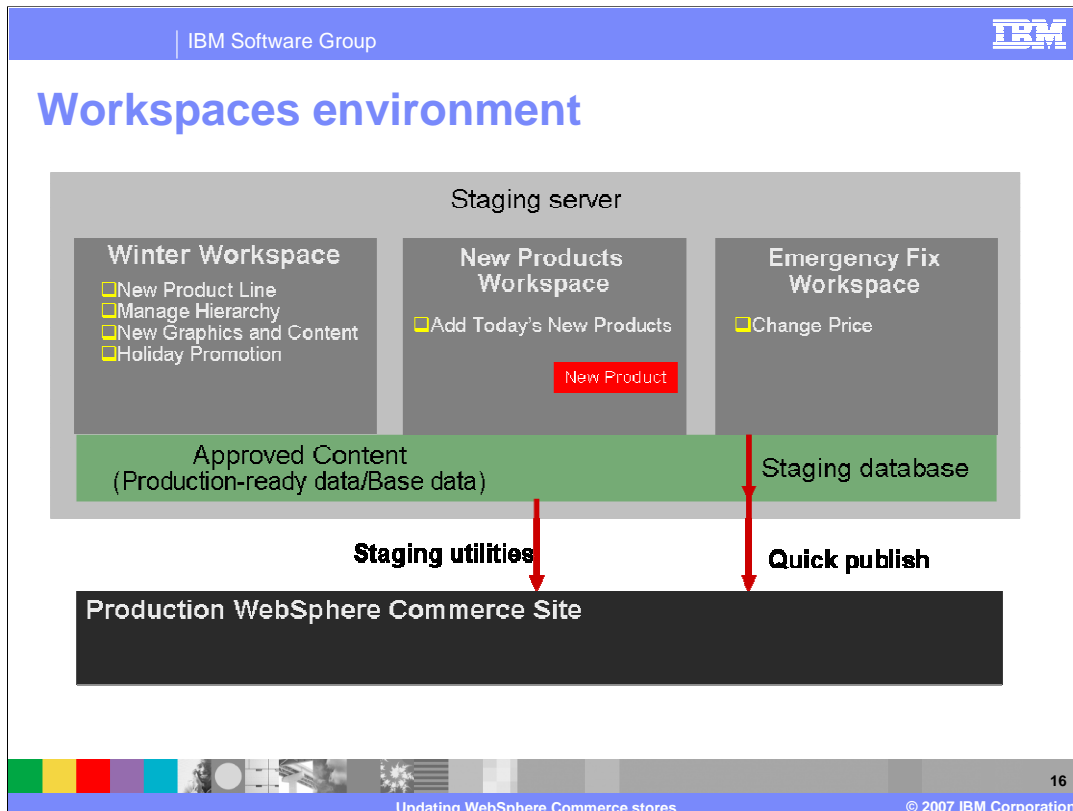
© 2007 IBM Corporation

Authoring servers and workspaces are available in WebSphere Commerce Professional and WebSphere Commerce Enterprise only.

WebSphere Commerce workspaces are isolated access-controlled work areas that act as private copies of the WebSphere Commerce database. Users authorized to access a workspace can make changes to the database in a workspace using the WebSphere Commerce tools without affecting the other data. The data in a workspace is separate from the data in other workspaces and the production-ready data.

Workspaces are divided into task groups and tasks. A task group is the smallest set of data to commit to production-ready data. A task is used to assign work to users.

The usage of the authoring server and workspaces gives business users control in a number of areas without having to involve the IT staff. For example, it allows to control all site changes in the areas of catalog, content and campaigns and to preview everything at the current time or in the future. It also provides a way to granularly schedule which changes should go to production and when. Business users can manage tasks, notifications, and approvals to help assign work and track work to completion. They can also manage multiple independent projects operating on unique timelines.



The WebSphere Commerce workspace environment provides the ability for business users to control and manage site changes to not just content but campaign information, promotions, targeted recommendation, site segmentation, and catalog and merchandising content.

This slide shows a customer running three workspaces in the workspace environment. The customer has site changes they are making for winter, site changes they are making on a recurring basis such as adding new products daily, and a workspace for making emergency changes.

In the winter workspace, this team of business users is creating a new product line for winter, updating catalog hierarchies, adding new graphics, promotional content, and creating a holiday promotion.

In the new products workspace, the same or different business users are updating products that load in from a back end system daily and making sure they ready for the production Web site.

In the emergency fix workspace, business users can make immediate changes to production to fix things such as price inaccuracies.

In the workspace environment, the green box is where all approved content resides. Once content is approved, it can be seen and used by users in any workspace. This scenario allows for a degree of interdependence across workspaces.

Workspaces commit and publish

- Scheduled commit
 - ▶ Schedule a date and time for changes to go to the production-ready data
- Regular commit
 - ▶ Commit occurs after all tasks in the task group are completed and, if required, the task group is approved
- Global publish
 - ▶ Production-ready data is published to the production server using the **stagingprop** utility, while managed files are published using the **fileprop** utility
- Quick publish
 - ▶ Commit occurs after all tasks are completed and, if necessary, approved. When the task group commit completes successfully, the data is published to the production server
 - ▶ This is a useful tool for correcting errors immediately
 - ▶ Quick publish target must be defined

You have several options for committing workspaces data to production-ready data and then publishing that data to the production server.

The **quick publish** target is normally defined during instance creation. However, if the production server was not yet available, that target will need to be updated before quick publishing can occur. This requires making changes to WebSphere Application Server configuration, the WebSphere Commerce configuration, and then re-starting the authoring server instance. If the publish to production server fails during a quick publish, the task group data remains committed to the production-ready data. The changes to the production-ready data from the task group commit can then be published to the production server using the stagingprop utility.

With a scheduled commit, Workspaces Administrators can select to schedule when the task group data is committed to the production-ready data. This allows the business team to complete all site campaign and content changes, review, and approve everything in advance and schedule the collection of changes to go to the production-ready data at a predetermined date and time.

With regular commit, task group data will be committed to production-ready data once tasks are completed and approved.

Workspaces locking

- Locking policies help prevent inconsistencies in workspaces data
 - ▶ Defines how changes are made to a managed asset
 - ▶ Defines who can make those changes
- Available policies are:
 - ▶ Workspaces
 - ▶ Task group (default)
 - ▶ Task
 - ▶ No locking
- Minimize commit failures by:
 - ▶ Activating task groups at particular times
 - ▶ Using scheduled promotion features

What happens if a change is made to the same product in two different task groups at the same time? This will depend on the workspaces locking policy. The locking policy controls how changes are made and who can make them.

The default policy is task group locking. With this policy, a managed asset is locked to the task group in which the asset is first changed. A user working in the same workspace but in a different task group can see the asset but cannot change the asset until the lock has been released. The lock is released when the task group is either completed or canceled.

The locking policy will not necessarily prevent users in other task groups from using the asset to create new relationships. For example, in Task Group A, Product X is created. In Task Group B, a user creates an upsell relationship between Product X and Product Y. If Task Group B is completed before Task Group A, a commit failure will occur because Product X does not yet exist in the production-ready data. This is not necessarily a problem. Task Group B will remain in a state where the work is complete and the system will attempt to commit the data again later.

Workspaces roles

- **Workspace manager**
 - ▶ Responsible for entire workspace
 - ▶ Creates workspaces, task groups, and tasks
 - ▶ Assigns task group approvers and content managers
- **Workspace task group approver**
 - ▶ Responsible for approving content created by content contributors
- **Workspace content contributor**
 - ▶ Responsible for completing assigned tasks
 - ▶ Uses the WebSphere commerce accelerator to complete the task

New roles have been added to WebSphere Commerce to support the functions of workspaces.

By default, a site administrator has all of the authorities of a workspace manager. However, it will be desirable to assign this role to someone separately.

By default, a category manager, marketing manager, and product manager have the authority of a content contributor. However, it might be desirable to assign this role to someone separately.

Task group approvers and content contributors must be explicitly assigned the role. These roles need to be added as assignable roles for your organization before you can give a user that role.

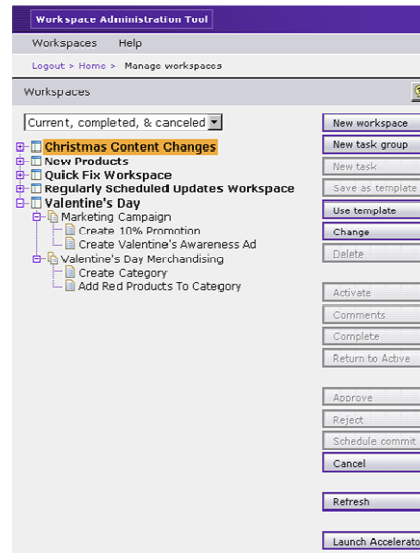
Task group approvals can also be done from WebSphere Commerce Accelerator. This gives you the flexibility to separate your approver role and your workspace administrator role – with the approver role not requiring access to the workspace administration tool.

Workspaces administration tool

- Used by business managers to to:
 - ▶ Create workspaces
 - ▶ Assign work
 - ▶ Track work to completion
 - ▶ Preview and approve changes
 - ▶ Reject / cancel changes
 - ▶ Schedule publishing

Accessed at:

https://<host_name>:8000/workspaceadmin



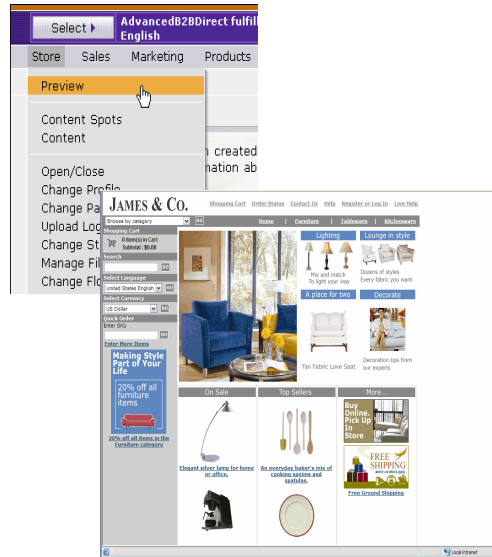
20

The workspace administration tool allows business users to create workspaces, assign work, track work, approve or reject changes, and schedule publishing.

The workspaces administration tool cannot be launched from the Windows program menu by default so you must type in the URL provided at the bottom of the slide.

WebSphere Commerce preview

- Preview all site changes working together
- Time and date driven preview to see how the site will look in the future
- Set inventory thresholds to see inventory driven product recommendations
- Preview in the context of a workspace or preview all approved change



Preview is another new feature in WebSphere Commerce V6 that allows business users to preview their changes in staging, authoring, and production environments. Preview is almost live. You can switch user context in the preview by logging into the store as any user you want.

Configure staging and authoring server

| | |
|------------|--|
| Instance | <input checked="" type="checkbox"/> Use staging server |
| Database | <input checked="" type="checkbox"/> Enable workspaces |
| Schema | |
| WebSphere | Read schema prefix: WCR |
| Languages | Write schema prefix: WCW |
| Web server | Workspaces: 5 |
| Staging | Production application name: WC_demo |
| | DataSource name: WebSphere Commerce DB2 Publish Da |
| | Database name: |
| | Database user name: |
| | Database user password: |
| | Confirm database user password: |

Staging and authoring servers are configured during instance creation. The Workspaces value is the maximum number of workspaces that can be created. This limit cannot be changed later. The Site Administrator is responsible for allocating sufficient schemas during instance configuration to manage the business needs of the site. The workspace managers must ensure that the number of simultaneous workspaces does not exceed the available pool of workspace schemas. Once a workspace is complete, the schema is released and can be re-used.

Updating applications with WebSphere Application Server

- WebSphere application Server administration console
 - ▶ Update entire EAR
 - ▶ Update individual modules or files
- Automate the process using:
 - ▶ wsadmin
 - ▶ Ant
 - ▶ Java management extensions (JMX)

Application update options

Full application
Select this option to replace the enterprise archive (*.ear) of the application. The uploaded enterprise archive replaces the existing application.

Single module
Select this option to update an existing module or to add a new module to the application. If the relative path to the module matches an existing path in the installed application, the uploaded module replaces the existing module. If the relative path to the module does not exist in the installed application, the module is added to the application.

Single file
Select this option to update an existing file or to add a new file to the application. If the relative path to the file matches an existing path to a file in the installed application, the uploaded file replaces the existing file. If the relative path to the file does not exist in the installed application, the uploaded file is added to the application.

Partial application
Select this option to update or add several files to an application. The files must be in a compressed file format such as .zip or .gzip. The compressed files are uploaded to the installed application directory. If the uploaded files exist in the same paths and file names, the uploaded files replace the existing files. If the uploaded files do not exist, the files are added to the application by specifying the relative path to the file.

Upload the archive file with the new or replacement files.

Local file system
Specify path:

Remote file system
Specify path:

Custom code needs to be deployed using the application update features of WebSphere Application Server. WebSphere Application Server provides several mechanisms for updating applications. It is a best practice to deploy WebSphere Commerce updates incrementally. You generally do not want to have to re-package and re-deploy the entire EAR every time you make a small change. The easiest method is to use the administration console.

Apache Ant is a popular Java-based build tool that uses XML configuration files. By combining various WebSphere Application Server tasks with those provided by Ant, you can create build scripts that compile, package, install, and test your application on the application server.

JMX is a framework that provides a standard way of exposing Java resources, for example, application servers, to a system management infrastructure. You can use the API to develop a program that automates the deployment process.

Deploying custom store updates

- | | | |
|---|---|---|
| <ul style="list-style-type: none"> ▪ Storefront assets <ul style="list-style-type: none"> ▶ JSPs ▶ HTML ▶ Images ▶ Properties files | → | <ul style="list-style-type: none"> ▪ Manually copy to file system ▪ Fileloader or Fileprop utility for managed files ▪ WebSphere Application Server application update |
| <ul style="list-style-type: none"> ▪ Business logic <ul style="list-style-type: none"> ▶ EJBs ▶ Command beans | → | <ul style="list-style-type: none"> ▪ WebSphere Application Server application update |
| <ul style="list-style-type: none"> ▪ Store data assets <ul style="list-style-type: none"> ▶ Registration of new or changed commands ▶ Access control policies ▶ Catalog updates ▶ Schema changes (new tables) | → | <ul style="list-style-type: none"> ▪ Loader package ▪ Staging utilities ▪ SQL |
| <ul style="list-style-type: none"> ▪ Other <ul style="list-style-type: none"> ▶ Configuration changes | → | <ul style="list-style-type: none"> ▪ Commerce Configuration Manager ▪ WebSphere Application Server administrative console ▪ Edit XML manually |

The developer will export the new store assets from the development environment. The deployment team and the development team have to decide how these assets should be packaged. They should also agree on who is responsible for ensuring that the directory structure of the exported assets matches that on the production system.

Recommended courses

Formal education exists for this product and you can find information on recommended training paths and certification tests at the these links.

- Application developer for WebSphere Commerce V6
<http://www-304.ibm.com/jct03001c/services/learning/ites.wss/us/en?pageType=page&c=a0011792>
- Business user for WebSphere Commerce V6
<http://www-304.ibm.com/jct03001c/services/learning/ites.wss/us/en?pageType=page&c=a0011793>
- System administrator for WebSphere Commerce V6
<http://www-304.ibm.com/jct03001c/services/learning/ites.wss/us/en?pageType=page&c=a0011794>



IBM provides these training paths for the skill or certification you want to explore.

Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject= Feedback about wcs60_UpdatingWebSphereCommercestores.ppt



You can help improve the quality of IBM Education Assistant content by providing feedback.

Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

WebSphere

Java, JMX, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2007. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.