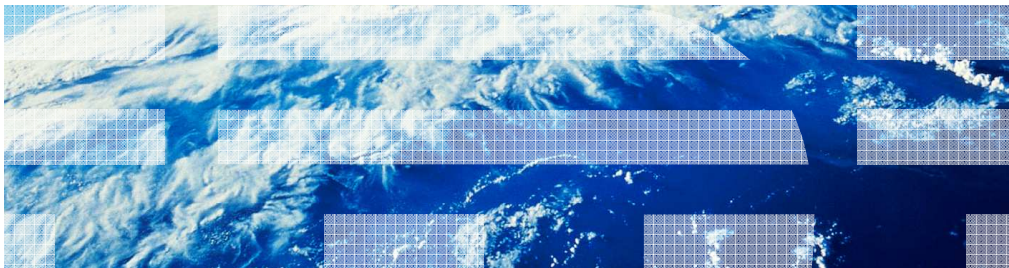# WebSphere Commerce Version 7 Feature Pack 3

Reducing total cost of implementation

This presentation provides an overview of the enhancements to reduce total cost of implementation in WebSphere Commerce Version 7 Feature Pack 3.

Table of contents

- Starter store companions
  - Review of starter store companion assets
  - Updates for Feature Pack 3
- New documentation framework

Reducing total cost of implementation © 2011 IBM Corporation

This presentation provides a review of the starter store companion assets and summarizes the updates available with Feature Pack 3. The presentation concludes with an overview of a new documentation framework in the WebSphere Commerce Information Center.

Section

Review of starter store companion assets

Reducing total cost of implementation

This section reviews the starter store companion assets.

WebSphere Commerce TCOI summary

Plan	Starter store use cases HTML assets package	Starter store site flow
Design	Starter store use cases Starter store site flow HTML assets package	Overall shopping flow diagram Order status transition flow diagram Order task command documentation
Develop	Starter stores Order task command documentation Refactored Order commands	WebSphere Commerce High Level Test Plan **Test** Starter store FVT Test Case Document Storefront Test Automation Engine and scripts
Operations	Data load utility WebSphere Commerce Build and Deployment tool	
Post-production	Generate HTML assets package	Legend Starter store companions

4   Reducing total cost of implementation   © 2011 IBM Corporation

This slide summarizes the various tools, assets, documentation and code that has been introduced in version 7 and version 7 feature packs to help reduce the total cost of implementation for a WebSphere Commerce solution. The areas highlighted in yellow are known collectively as the starter store companion assets. These assets are used along with the starter store code to help you quickly get a store up and running. The individual assets and the enhancements in Feature Pack 3 are summarized in this presentation.

Starter store use cases

- Describes main flows, alternate flows and exception flows for all starter store pages

- Provides store flow summary diagrams

- Organized by store area
    – Registration
    – My Account
    – Catalog browsing
    – Shopping cart
    – Checkout
    – Store locator

Reducing total cost of implementation

The starter store use cases document describes the main, alternate and exception flows for both the Madisons and Elite starter stores. Use this document as a starting point for defining your own store use cases. You can remove use cases for features you're not using and add use cases for new functionality.

HTML assets package

- Direct HTML editing for starter stores
- Allows web designers to work with HTML / CSS
- JSP highlighting to help store developers to map HTML changes to store JSP files
- HTML capture done with Storefront Test Automation Engine
  – Already done for Madisons and Elite
  – Can be configured and run against any store

Reducing total cost of implementation

The static HTML asset helps store designers and developers work together more efficiently. It can also be useful for doing a fit-gap analysis or rapid prototyping. This set of HTML, CSS and JavaScript assets allows you to walk through an entire Madisons and Elite store flows without having a full WebSphere Commerce environment running. web designers can take the existing starter store pages and, using only HTML and CSS, create a custom version of the pages for store developers to build. Store developers can then make use of the JSP highlighting feature to easily see which JavaServer Page files to update and transfer the styling changes to the real store JSP pages.

Once you have a customized store, you can create a new static HTML package using the Storefront Test Automation Engine. Using automated test scripts, you can capture an updated version of the store HTML. Additional utilities are provided to help you convert the captured HTML into a new static package ready for your next store update.

Highlight tool

CTRL + ALT + H to activate

JSP name

You can use the highlight feature on the static page using CTRL-ALT-H and it will mark up the page with borders and give you the JavaServer Page files names.

High Level Functional Verification Test Plan Template document

- Agreements between project management team and developers on the test scope

- Microsoft Word document format

- A document template, not considered to be a complete plan

- No changes in feature pack 3

The High Level Functional Verification Test Plan document is an agreement between the project management team and developers on test coverage and scope. The document is a template that is not considered to be complete. In the document, instructions and samples are provided to help you complete your own test plan. A screen capture in this slide shows part of the contents in the document.

Functional Verification Test Case Document

- Provides detailed steps to test the features contained in the Madisons and Elite starter stores

- Modifiable Microsoft Excel document

- Contains the set of test cases that align with the use case documentation

- Contains the test data to run the test cases

- Use it as starting point to extend to create your own test case document

　　　Reducing total cost of implementation　　　© 2011 IBM Corporation

The Functional Verification Test Case Document provides the detailed test procedures or steps to test the features contained in the Madisons and Elite starter stores. It is a customizable test plan in Microsoft Excel format. The test cases provided in this document align with the use case documentation. You can start with this test case document and modify it to accommodate your customizations made to the starter stores during the implementation phase.

Functional Verification Test Case Document - example

	A	B	C	D
1		**SCENARIO TO USE CASES MAPPING**		
2				
3		*<SCENARIO TEMPLATE CAN BE FOUND AT THE END. Scenario 49 and 50>*		
4				
5				
6		**Scenario 1**	**Scenario Objective**	
7		FSTORE.B2C.01	Register a Shopper	TEST CONDITION MATRIX
8			USE CASE TO SCENARIOS MAP	TEST CASES
9		**#**	**Use Case(s)**	**Additional Document Link**
10		1	101 - Shopper registers for an account with the store	http://publib.boulder.ibm.com/infocenter/wchelp/v7r0m0/index.jsp?topic=/com.ibm.commerce.madisons-starterstore.doc/refs/rsmmadisonpregister.htm
11		2	401 – Shopper adds an item to the Shopping Cart	http://publib.boulder.ibm.com/infocenter/wchelp/v7r0m0/topic/com.ibm.commerce.madisons-starterstore.doc/refs/rsmmadisonpshoppingcart.htm
12		3	405 – Shopper views the content of the Shopping Cart	http://publib.boulder.ibm.com/infocenter/wchelp/v7r0m0/topic/com.ibm.commerce.madisons-starterstore.doc/refs/rsmmadisonpshoppingcart.htm
13		4	501 – Guest shopper checks out	http://publib.boulder.ibm.com/infocenter/wchelp/v7r0m0/topic/com.ibm.commerce.madisons-starterstore.doc/refs/rsmmadisoncheckoutflow.htm
14		5		
15		6		
16		7		
17		8		
18		9		
19		10		
20				
21				
22		**Scenario 2**	**Scenario Objective**	
23		FSTORE.B2C.02	Shopper Signs in to the Store	TEST CONDITION MATRIX
24			USE CASE TO SCENARIOS MAP	TEST CASES
25		**#**	**Use Case(s)**	**Additional Document Link**
26		1	102 – Registered shopper signs in to the store	http://publib.boulder.ibm.com/infocenter/wchelp/v7r0m0/topic/com.ibm.commerce.madisons-starterstore.doc/refs/rsmmadisonpsignin.htm
27		2	103 – Registered shopper signs out of the store	

TestOverview / GeneralSetup / UsecaseScenarioMap / **ScenarioUsecaseMap** / TestCondMatrix / Testcases / Data / Messages / Accessibility / Globalization /

This screen capture shows an example of one worksheet in the test case document. The selected worksheet provides a mapping for each test scenario back to the use cases it covers. You can see several other tabs along the bottom that include information such as a use case to test scenario mapping, the individual test cases and the test case data.
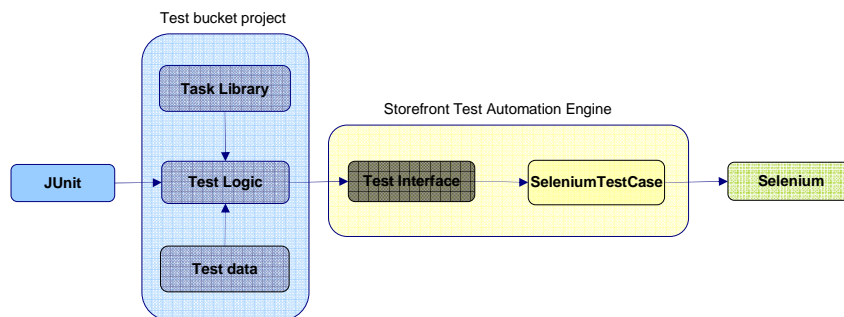
Storefront automation harness is provided to assist in automating functional verification testing. It consists of four key assets: JUnit, test bucket project, Storefront Test Automation Engine and Selenium.

JUnit is a Java tool used to run test cases. It is used by the storefront automation harness to initiate tests.

The test bucket project is a Java project which contains test logic, test data and task library packages.

The test logic package contains test scripts. A test script is a Java class which contains all the logic for a test scenario.

The test data package contains data files which hold the actual data for each test script.

The task library package contains the reusable tasks. A library is a Java class. A task is a Java method which implements a storefront action, such as clicking on the header link.

Storefront Test Automation Engine is a Java project which contains the code to interact with the Selenium test tool to do the test. Two key packages are included in this project: Test interface and Selenium Test case.

Test interface provides a set of methods, such as click, select, and type. The test scripts use these methods to instruct the Selenium test tool to perform the corresponding action in the actual browser.

Selenium Test case is the implementation of test interface for Selenium tool.

Test scripts

- A Java class that automates a single test scenario
- Contains one or more JUnit test case methods
- Test cases perform storefront actions by calling the tasks in a specific order
- Tasks use constants to represent storefront object DOM IDs
- Tasks call the test interface action methods to send action to browser
- Test interface actions use DOM IDs passed by tasks to access the storefront objects
- Each test script references one test data file

```
                         Test scripts

        Call          Call                    Reference

    Tasks          Test interface action     Test data
  (contained in                            (contained in
  task libraries)                          data files)

        Use

    Constants
  (contained in
  constant files)
```

12   Reducing total cost of implementation   © 2011 IBM Corporation
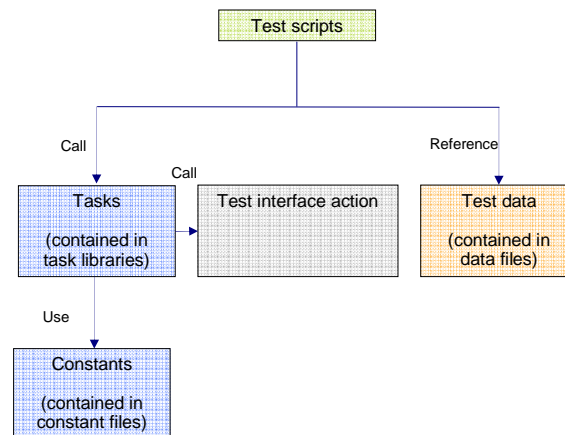
A test script is a JUnit Java class that automates a single test scenario. Each test script contains one or more test cases. A test case is a Junit test case method which holds the logic to perform storefront actions by combining tasks from the task libraries in a specific order. The tasks are contained in task libraries. The tasks call the test interface action methods to send actions to the browser. They pass values, stored as constants, to represent the DOM ID of the page element to take the action on. The test interface uses the DOM ID to access the storefront object. The data used for a test case is from a data file. You should have one data file for each test script.

Section

Updates for Feature Pack 3

13   Reducing total cost of implementation   © 2011 IBM Corporation

This section summarizes the updates to starter store companion assets in Feature Pack 3.

Starter Store Companion updates

- Use case and static HTML updates
  – Base starter stores
  – Base starter stores + store enhancements
  – Madisons + store enhancements + social commerce
  – Brazil starter store (post GA for use cases)
  – China starter store

- Test case updates
  – Search
  – Social Commerce (post GA)

- Test script updates
  – Search (post GA)

Reducing total cost of implementation

All of the existing starter store companion assets have been updated to the feature pack 3 level. In addition, some features from the add-on store archives have been included. This slide summarizes the features included in the latest version of each asset. The use case document and static HTML package is available in five different versions. Each version contains a different subset of features. You can choose the version that most closely matches the features you plan to use in your store. The test case document has been updated to include test scenarios for the search and social commerce features. The next version of the automated test scripts will include test cases for the WebSphere Commerce search feature.

Use case organization overview

- Top level use case represents a unique shopper function
  - Examples: search, create a blog entry

- Alternate flow for a use case represents an optional way to achieve the same function
  - Examples: Ajax versus regular checkout, various add to cart methods

- To customize
  - Locate all use cases that deal with the store function being changed
    - Might span multiple shopper function use cases
    - Refer to the use case's **Store Function** and **Related Use Cases**
  - Update the affected use cases

Reducing total cost of implementation © 2011 IBM Corporation

The use case document is organized based on shopper functions within the store. For each shopper function, there might be alternate ways to complete the same task. These are recorded as alternate flows for a use case. When it comes to customization, you might need to update or remove use cases related to a specific store function, such as subscriptions, that spans multiple use cases. You can use the Store Function and Related Use Cases fields in the use case definition to locate the other use cases to be modified.

Section

New documentation framework

Reducing total cost of implementation

This section introduces a new documentation framework.

TCOI documentation framework

- Problem
  - Can be time consuming to find the right information when you need it
  - Easy to get lost in the details
- Solution
  - Create a central summary of implementation resources
    - Grouped by project phase
    - Dashboard experience
    - Consolidated from multiple sources

Reducing total cost of implementation

Having access to the right information at the right time is an important part of reducing the cost of implementation during a project. To help ensure you can quickly find the right assets and information for each phase of your project, a new documentation framework has been added in the WebSphere Commerce Information Center. The framework groups information by project phase and introduces a new dashboard style display method. It also consolidates information from multiple sources to provide a summary of all available implementation resources. You can find this new information resource in the Planning book in the Information Center.

Planning book

Concept stage

During the Concept stage of your project you communicate WebSphere Commerce capabili...
WebSphere Commerce provides resources that you can use to demonstrate, describe and...
capabilities.

Reference material	Use this to ...
HTML Store	WebSphere Commerce provides a set of static, navigable HTML pages f... the store pages and their different supported flows. For example, there shopping cart, a shopping cart page with more than 20 items and pagina opening the HTML files in a browser. You do not need a development en server. While viewing a static page you can display the names of the JS highlight feature. The highlight feature marks off each JSP page and sho...
Use case documentation	During a customization project, you must perform architectural design e store. To reduce the time required to document use cases for your stor clearly describes the functionality provided with a starter store.

This slide shows a screen capture of the new Planning book. Each project phase has its own tab so you can easily navigate between phases without returning to the table of contents on the left. The table on each tab provides links to the reference material available for the selected phase along with a description of when and how to make use of each resource.

Summary

- Starter store companions
  - Review of starter store companion assets
  - Updates for feature pack 3
- New documentation framework

　　Reducing total cost of implementation　　

This presentation provided a review of the starter store companion assets and summarized the updates available with feature pack 3. The presentation concluded with an overview of a new documentation framework in the WebSphere Commerce Information Center.

References

- Testing starter stores
    - http://publib.boulder.ibm.com/infocenter/wchelp/v7r0m0/topic/com.ibm.commerce.starterstores.doc/refs/rsmstorefrontautomationharness.htm

- Static starter stores
    - http://publib.boulder.ibm.com/infocenter/wchelp/v7r0m0/topic/com.ibm.commerce.starterstores.doc/refs/rsmstatichtmlassets.htm

- Planning book
    - http://publib.boulder.ibm.com/infocenter/wchelp/v7r0m0/topic/com.ibm.commerce.planning.doc/concepts/cpl_planning_main.htm

Reducing total cost of implementation

This slide provides some reference links.

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 email feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_TCOI.ppt

This module is also available in PDF format at: ../TCOI.pdf

21   Reducing total cost of implementation   © 2011 IBM Corporation

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

Trademarks, disclaimer, and copyright information

IBM, the IBM logo, ibm.com, and WebSphere are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of other IBM trademarks is available on the web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY.
Java, and all Java-based trademarks and logos are trademarks of Oracle and/or its affiliates.

Other company, product, or service names may be trademarks or service marks of others.

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS OR SOFTWARE.