



IBM Software Group

## WebSphere® Commerce Feature Pack 2

*Web 2.0 Store Model*

*Overview*



@business on demand.

© 2007 IBM Corporation  
Updated May 7, 2007

Welcome to the WebSphere Commerce Feature Pack 2 Web 2.0 Store Model Overview presentation.

## Agenda

- Web 2.0 evolution of the Internet
- WebSphere Commerce Web 2.0 feature
- AJAX overview
- Installing Web 2.0 reference application



The agenda for this presentation is to discuss the following topics:

The evolution of the internet into the business model called Web 2.0, an overview of the Asynchronous JavaScript and XML technology called AJAX, the specifics of the WebSphere Commerce feature called Web 2.0, and specifics on installing the Web 2.0 reference application.

## Section

# ***Web 2.0 evolution of the Internet***

This section discusses the Web 2.0 internet evolution.

## Web 2.0 defined

- Evolution of the World Wide Web from first generation to second generation
  - ▶ Web 2.0 coined October 2004 O'Reilly Media sponsored conference
  - ▶ According to Tim O'Reilly, 2001 dot-com bubble burst marked turning point in doing business on the Web
- Conceptual shift assisted by specific technologies
  - ▶ Community intelligence instead of individual authority
  - ▶ Rich and responsive but simple user interfaces
  - ▶ Software as a service instead of a product

4

Web 2.0 Store Model overview

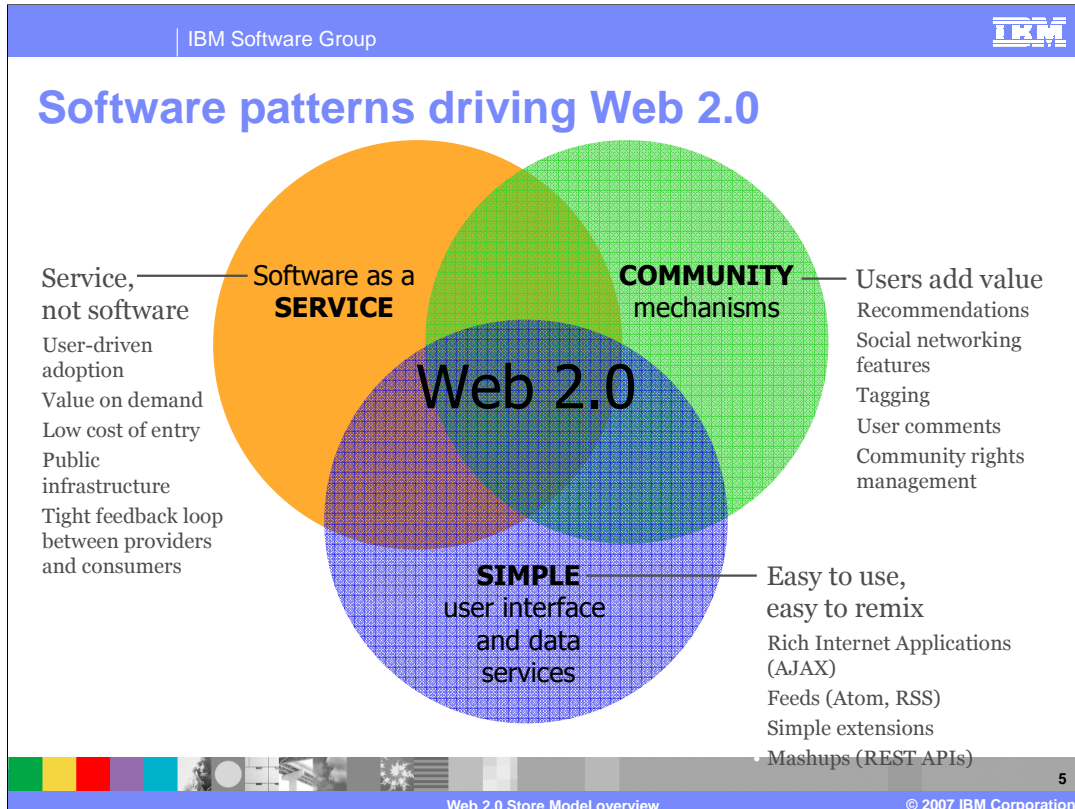
© 2007 IBM Corporation

The term “Web 2.0” was first coined in October, 2004 at an industry conference sponsored by O'Reilly Media. According to Tim O'Reilly, the dot-com bubble burst marked a turning point in doing business on the Web. Since 2001, there has been a conceptual shift in how business is done on the Web. This conceptual shift has many different facets but this presentation only talks about three facets:

Community intelligence has more power than the authority of an individual.

Rich and responsive, but simple user interfaces gain acceptance more rapidly than slower or more complicated interfaces.

Software is made available as a service to use for a limited time rather than as a product to own.



There are three broad software patterns associated with Web 2.0:

The first is **software as a service**. It enables you to adopt capabilities directly, and gain immediate benefits from it. Many Web 2.0 offerings have a very low cost of entry – another distinguishing characteristic that levels the playing field for small businesses. And of course they use the public infrastructure, so anyone in the world can walk up and use it with a Web browser.

The second key pattern is the use of **community mechanisms**. Most Web 2.0 businesses use one or another kind of mechanism to enable you to play a part in the service, adding value as they use it.

The third pattern is the **simplicity of the user experience** and the various interfaces by which developers can access data and capabilities. There are significant improvements in user interface design and responsiveness based on AJAX methods [Asynchronous JavaScript and XML]. Lots of simple and highly scalable mechanisms have emerged, such as feeds, simple extension mechanisms, and well-behaved HTTP-based APIs. Developers are building all kinds of situational applications on top of services, remixing them in various ways, without ever having to contact the service providers.

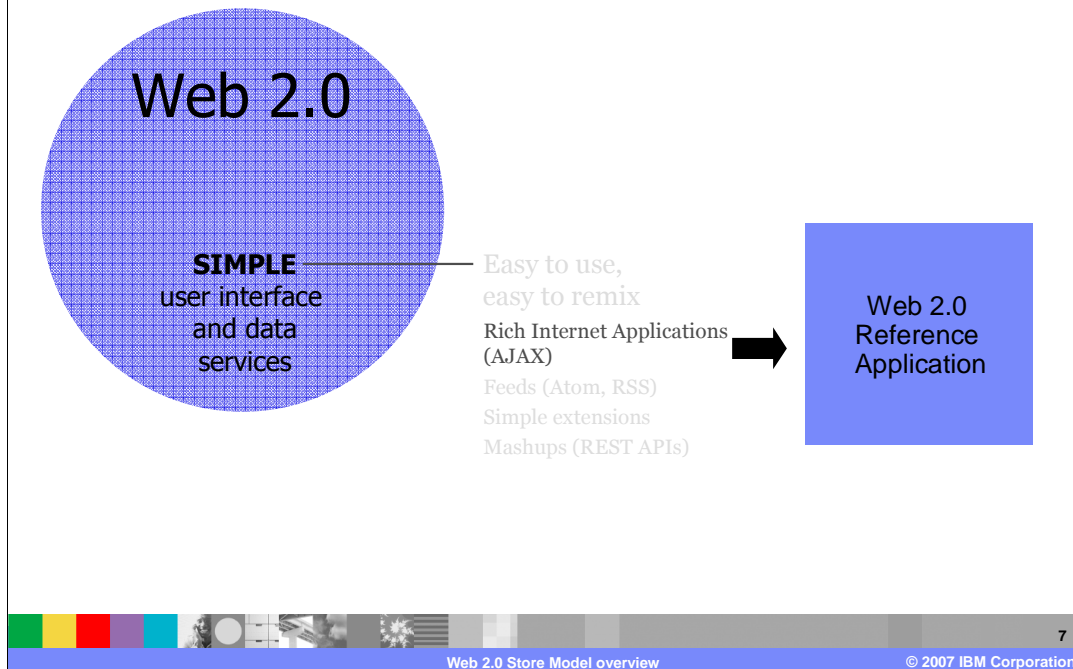
## Web 2.0 examples

Flickr	Community User interface Service
Blogger	Community
Wikipedia	Service Community
Google AdSense	Service
MySpace	Community
Google Earth	Service User interface

6

The table above gives some examples of modern Web sites that exhibit one or more of the conceptual facets. The table shows examples of the Web applications Flickr, Blogger, Wikipedia, Google AdSense, MySpace, and Google Earth.

## WebSphere Commerce Web 2.0 feature



The WebSphere Commerce Web 2.0 feature falls into the category of simple user interface. The feature is implemented as a reference application that implements a rich and responsive user interface using the AJAX programming model. The AJAX programming model is discussed in more depth later in the presentation.

## Section

# ***WebSphere Commerce Web 2.0 Store Solution feature***

This section discusses the specifics of the WebSphere Commerce Web 2.0 Store Solution feature.



## Web 2.0 reference application

- Reference application provides
  - ▶ Starter store and catalog store archives
  - ▶ User interface widget files
- Download reference application from  
[https://www14.software.ibm.com/webapp/iwm/web/preLogin.do?source=swg-wc&S\\_PKG=custp](https://www14.software.ibm.com/webapp/iwm/web/preLogin.do?source=swg-wc&S_PKG=custp)
- Reference application implemented using AJAX technology



The WebSphere Commerce Web 2.0 feature is implemented as a reference application. The reference application provides a starter store with catalog packaged in two store archives, and a collection of user interface widgets. The reference application can be downloaded from the URL above.

## Web 2.0 enhanced user interface

- Product fast finder
- Product quick view
- Interactive marketing spot
- Single page checkout
- Drag and drop page objects



The reference application implements several enhanced user interface elements. A product fast finder, product quick view, interactive marketing spot, and a streamlined single page checkout are provided. These elements use a drag-and-drop interface to perform object selection.

## Product fast finder

- Reducing selection criteria reduces result set

The screenshot illustrates a product fast finder interface for coffee makers. It shows two stages of filtering:

**Stage 1: Initial Search**

- Price Range:** \$10 to \$200 (slider)
- Brands:**  Sharpson,  AromaStar,  Enzi,  Kitchen's Best
- Features:**  Capacity,  Timer,  Pause 'n Serve,  Auto-off,  Grinder,  Espresso
- Results:** 23 products meet your criteria (out of 25). Displaying products 1 - 12 out of 23.

**Stage 2: Filtered Results**

- Price Range:** \$10 to \$200 (slider)
- Brands:**  Sharpson,  AromaStar,  Enzi,  Kitchen's Best
- Features:**  Capacity,  Timer,  Pause 'n Serve,  Auto-off,  Grinder,  Espresso
- Results:** 6 products meet your criteria (out of 25). Displaying products 1 - 6 out of 6.

The filtered products shown are:

- Sharpson Aroma Express 10 cup Coffee Maker \$19.99
- Kitchen's Best 8 cup Coffee Maker \$29.99
- Kitchen's Best 8 cup Drip Coffee Maker \$29.99
- Kitchen's Best Stay or Go Coffee Maker \$89.99
- Kitchen's Best Digital 12 cup Coffee Maker, Red \$99.99
- Kitchen's Best Digital 12 cup Coffee Maker, Green \$99.99

© 2007 IBM Corporation

A product fast finder allows a shopper to quickly narrow a product search by selecting or deselecting product attributes. A product price range selector has an upper and lower limit slider which limits the product selection to be between the upper and lower bounds.

## Product quick view

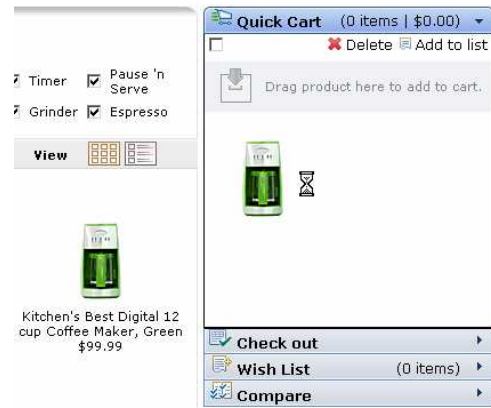
- Hovering over image pops product detail window



A product quick view shows a product summary pop-up screen when the pointer hovers over the item.

## Drag-and-drop to cart

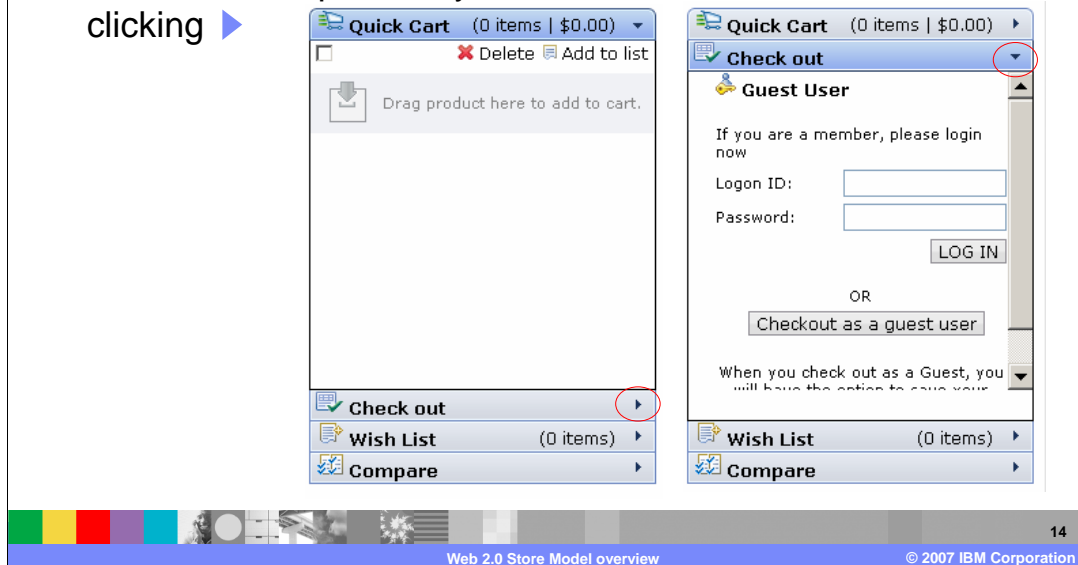
- Drag item to cart



Items from the product selection area can be selected for purchase by dragging them to the shopping cart area.

## Quick checkout

- Multipanel accordion window slides panels by clicking ▶



Quick checkout allows you to quickly transition from the shopping cart to a compact checkout area without refreshing the screen.

## Single page checkout

**CheckOut** | 1 Shipment | 2

Address: Eric Rose | 1245 Juniper Ct | Brookwood, MA 01903 (Edit)

Method:  Regular Delivery |  Specialty Advanced Shipping Options

Shipping instructions:  Required shipping date: Year:  Month:  Day:  (YYYY MM DD)

Items	QTY	Price
Shannon Deluxe 10 Cup Digital Coffee Maker SKU: SA-0261	1	\$79.99
Kitchen's Best 2.5ea.Dig. Coffee Maker SKU: SA-0546	1	\$38.00

Address: Eric Rose | 1245 Juniper Ct | Brookwood, MA 01903 (Edit)

Method:  Regular Delivery |  Specialty Advanced Shipping Options

Items	QTY	Price
Shannon Deluxe 10 Cup Digital Coffee Maker SKU: SA-0261	1	\$79.99
Kitchen's Best 2.5ea.Dig. Coffee Maker SKU: SA-0546	1	\$38.00

Payment Method:  Payment Method

\*Pay By:  Select

Coupons and Promotions	
Order Subtotal:	\$95.99
Discount Adjustment:	\$0.00
Burcharge Adjustment:	\$0.00
Tax:	\$0.00
Shipping Tax:	\$0.00
<b>ORDER TOTAL:</b>	<b>\$69.08</b>

Shipping & Returns info: Shipping, Order & Status, Our Return Policy, Contact Us, Store

Quick Cart: Kitchen's Best 2.5ea. Dig. Coffee Maker \$38.00, Shannon Deluxe 10 Cup Digital Coffee Maker \$79.99, Kitchen's Best 2.5ea. Dig. Coffee Maker \$38.99, Shannon Deluxe 10 Cup Digital Coffee Maker \$79.99

We also suggest:

Single page checkout allows the shopper accomplish all checkout steps on one page.

## Interactive marketing spot

- Promotion area with scrolling ticker of featured products



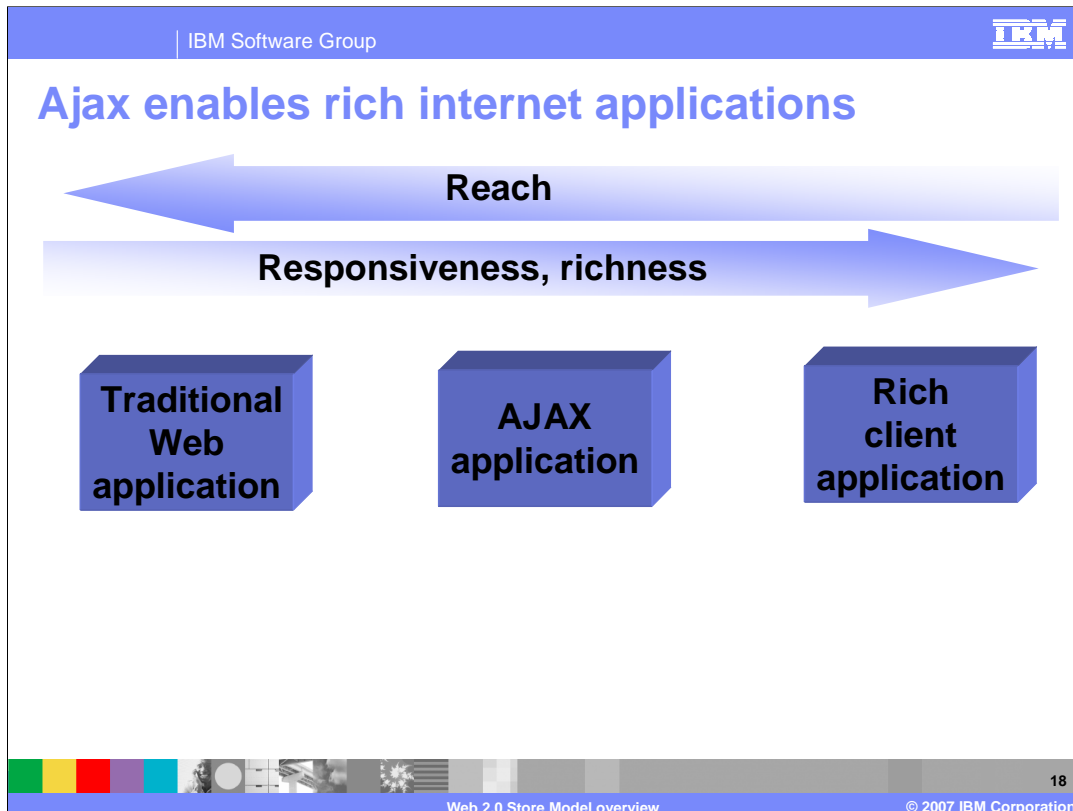
The interactive marketing spot shows a scrolling ticker of featured products. The shopper can scroll and pause the ticker with scroll controls.



## Section

# ***AJAX overview***

This section discusses an overview of the AJAX programming model.



Ajax is a fundamental technology pattern for supporting rich internet Web2.0 applications with recent widespread industry adoption.

Traditional Web applications get the most user reach because all browsers can run traditional Web applications. Rich client applications supply the most responsive and rich feature set, but these applications do not run in a browser. They typically run in a Java virtual machine and therefore require more resources to run. AJAX applications fit in the gap between traditional Web applications and rich client applications. AJAX can run on most modern browsers, but may not run on all browsers. AJAX can provide a moderately rich user experience above that of traditional applications.

## Traditional Web application interaction



Click

Browser makes HTTP request

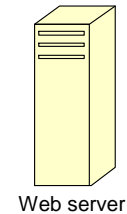
Wait



Server returns a **full** new page

Browser renders full page

Ready



A traditional Web application interacts with you when you click on some object or link on the page displayed in the browser.

Upon clicking, you begin waiting for something to happen.

The browser makes an HTTP request to the server. The server returns a full new page and the browser then renders that page.

When the page is done rendering, the wait is over and you can begin examining the new page.

This wait can take several seconds.

## AJAX application interaction



Click

Client makes HTTP request

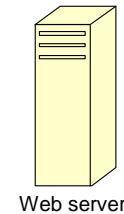
Wait

Server returns partial page info

Browser renders partial page

Ready

More requests are served asynchronously without user awareness



Web server

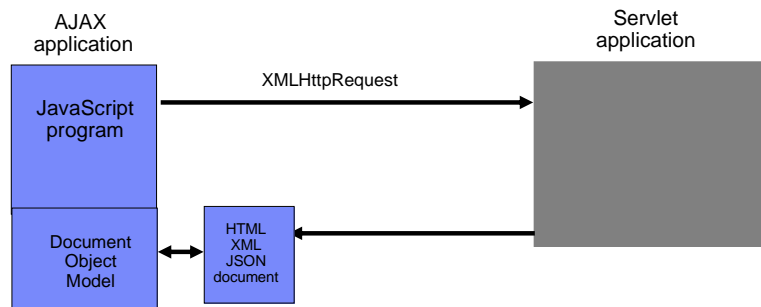


An AJAX application uses a more efficient user interaction model. When you click an object on the page, the browser makes a request for a part of a page, not a full page. The amount of data returned is smaller which takes less time to transmit. The browser renders less data and your wait time is less. A more important aspect of the interaction model is the browser can request data asynchronously, that is, independent of clicking an object. The data can be fetched ahead of time, anticipating user requests. When you eventually make an action, the data has already stored in the browser and can be rendered without waiting for transmission from the server.

This asynchronous interaction model results in a responsive feel to the application.

## Ajax programming model

- New way to use existing Web technologies
  - ▶ JavaScript
  - ▶ XML
  - ▶ HTML
  - ▶ CSS



Ajax programming is a new way to use existing Web technologies. An AJAX program consists of a collection of HTML and Cascading Style Sheets to describe the page layout. JavaScript programs loaded with the page content makes the page dynamic. A JavaScript program requests data from a servlet running on a server using the XMLHttpRequest function. The servlet returns a document that represents the internal structure of the page know as the Document Object Model (DOM). The return document can be in either HTML, XML, or JSON formatted data. JSON stands for JavaScript Object Notation, and will be discussed in more detail in a subsequent presentation on Web 2.0 programming model. The DOM has a programming interface accessible through JavaScript which interprets the returned document and uses the information to update a subset of objects on the page.

## AJAX browser requirements

- Modern Web browser that supports JavaScript, DOM, and XMLHttpRequest
  - ▶ JavaScript must be enabled through preferences
- No plug-ins required

AJAX can run on most modern browsers developed within approximately the past three years. The browser must be capable of running JavaScript, the Document Object Model (DOM) and the XMLHttpRequest JavaScript function. The browser preferences must also allow JavaScript programs to operate. Since almost all modern browsers support JavaScript natively, no additional plug-ins are required.

## AJAX advantages

- Improved response
  - ▶ Less data transmission and rendering
  - ▶ Background data requests means no staring at a blank screen
- Rich user interface features using external toolkits
  - ▶ Drag-and-drop
  - ▶ Animation
  - ▶ Dynamic page layout containers
  - ▶ Advanced controls such as sliders, accordions, tabbed palettes

AJAX has the potential to improve the responsiveness of applications because less data is transmitted and rendered. Since data is requested asynchronously, you spend less time staring at a frozen screen waiting for the next page to load. Rich user interface features such as drag-and-drop, dynamic page layout containers, sliders, and tabbed palettes are enabled using external toolkits.

## AJAX disadvantages

- Difficulty finding information on the page
  - ▶ Location of updated information not apparent
  - ▶ Unexpected loss of focus for page elements
- Browser history bypassed
  - ▶ Bookmarking is limited
  - ▶ Browser back and forward is limited
- Good AJAX programming practices can mitigate these disadvantages

AJAX programs have some disadvantages. Since parts of the page can be updated asynchronously to user interactions, users may find it difficult to find where the changed information appears on the page. Sometimes the pointer focus is lost because a page object has changed. Since AJAX programs typically do not do full page fetches, the browser page history is effectively bypassed, making the browser forward and back buttons not functional. For the same reason, bookmarking of pages is limited because the browser's idea of the page URL does not change. You will see in a subsequent presentation on AJAX programming best practices on how such disadvantages can be mitigated by good programming practices.



## Dojo toolkit

- Open source collection of JavaScript libraries
  - ▶ Drag-and-drop
  - ▶ Standard widgets for buttons, menus, accordion containers, dialogs
- JavaScript data structures
- Event driven programming model
- WebSphere Commerce Web 2.0 reference application uses Dojo toolkit 0.4.1
- For more information and downloads  
<http://dojotoolkit.org>

AJAX programs often use toolkits to make programming easier. The Web 2.0 reference application uses the Dojo toolkit. The Dojo toolkit is an open source collection of JavaScript libraries which aggregate user interface objects, called widgets. Widgets implement rich user interface components such as drag-and-drop, buttons, menus, and screen area containers. The Dojo toolkit also includes JavaScript data structures, and an event driven programming model. The Web 2.0 reference application uses the Dojo toolkit 0.4.1. For more information and toolkit downloads, see the URL in the slide.

## Section

# ***Installing Web 2.0 reference application***

This section discusses how to install the Web 2.0 store solution reference application.

## Web 2.0 reference application prerequisites

- Feature pack 2 installed and operational
- Component services feature enabled
- Browser minimum product level
  - ▶ Microsoft® Internet Explorer® 6.0
  - ▶ Mozilla FireFox 1.5
  - ▶ Opera 8.5
  - ▶ Apple Safari 2.0
  - ▶ Netscape 8.0
- Dojo toolkit 0.4.1 minimal package

The Web 2.0 reference application has these prerequisites:

1. Feature pack 2 is installed
2. Component services feature is enabled.
3. Browser minimum levels are Microsoft Internet Explorer 6.0, Mozilla FireFox 1.5, Opera 8.5, Apple Safari 2.0, and Netscape 8.0.
4. Dojo toolkit 0.4.1 is the minimal package.

## Steps to install reference application

- Download reference application from [https://www14.software.ibm.com/webapp/iwm/web/reg/download.do?source=swg-wc&S\\_PKG=custp&lang=en\\_US&cp=UTF-8](https://www14.software.ibm.com/webapp/iwm/web/reg/download.do?source=swg-wc&S_PKG=custp&lang=en_US&cp=UTF-8)
- Download Dojo toolkit
  - ▶ dojo-0.4.1-minimal.zip from dojotoolkit.org
- Enable component services
- Merge IBM widgets and Dojo widgets into WC application archive
- Copy store archives to correct directories
- Publish starter store

28

Web 2.0 Store Model overview

© 2007 IBM Corporation

The process to install the Web 2.0 reference application consist of these steps:

First download the reference application from the URL in the slide. Next download the Dojo toolkit from dojotoolkit.org. Select the download archives link and look for the file named in the slide. Next enable component services. Then merge the IBM widgets with the Dojo widgets and add them to the WC enterprise application. Then copy the provided store archive files to the correct directories and publish the starter store.

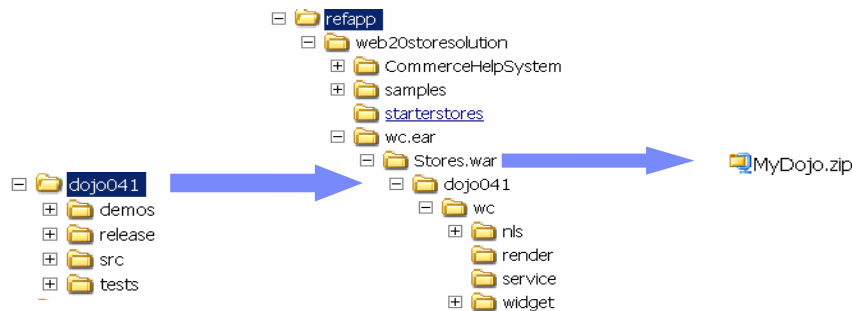
Subsequent slides provide you more details on how to do these steps.

## Installing reference application details

- Enable component services
  - ▶ Server environment
 

```
config_ant -buildfile <WCS_HOME>/components/common/xml/enableFeature.xml -DinstanceName=<instance> -DfeatureName=component-services -DdbUserPassword=dbUserPassword
```
  - ▶ Developer environment
 

```
enableFeature -DfeatureName=component-services
```
- Merge IBM widgets and Dojo widgets into WC application
  - ▶ Combine dojo .zip file and IBM widget directory



Enable component services for the server environment with the `config_ant` command. A command example is given in the slide. Enable component services for the Developer environment with the `enableFeature` command. A command example is given in the slide. Merging the IBM widgets with the Dojo widgets can be accomplished many ways. This is the suggested way. First unzip both the Dojo toolkit and Web 2.0 reference app. Rename the widget directory to `dojo041` to match that of the IBM provided widgets. Copy the former to the latter using a graphical file manager such as Windows Explorer. Create a .zip file of the latter at the directory `Stores.war`. More explicit detailed instructions are provided in the documentation accompanying the reference application.

## Installing reference application details

- Update WC\_instance.ear with merged file
  - ▶ WebSphere Application Server administration console partial application update

Partial application

Select this option to update or add several files to an application. Use a valid compressed file format such as .zip or .gzip. The compressed file is unzipped into the installed application directory. If the uploaded files exist in the application with 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. You can remove existing files from the installed application by specifying metadata in the compressed file.

**Upload the archive file with the new or replacement files.**

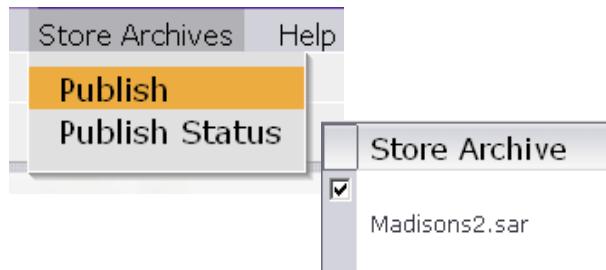
Local file system

Specify path

Update the WC enterprise application using the WebSphere Application Server administration console. Select the WC application and select update. Then select partial update and specify the .zip file generated in the previous step.

## Installing reference application details

- Copy starter store files
  - ▶ Copy Madisons2.sar Madisons2SampleData.sar to <WCS\_HOME>/instances/<instance name>/sar
- Publish Madisons store



Copy the 2 starter store files to the sar directory under the WebSphere Commerce instance directory. The exact directory path and file names are shown in the slide. Then publish the starter store using the WebSphere Commerce administration console. Select Store Archives > Publish. On the next screen select the Madisons2.sar archive.

## Problem determination

- Logs to gather
  - ▶ Server runtime log – SystemOut.log in instance profile  
<WAS\_HOME>/profiles/<instance>/logs/server1
  - ▶ Feature enable logs in instance logs directory  
<WC\_HOME>/instances/<instance>/logs

Problem determination data can be gathered from the server runtime log in the server profile logs for server1. The slide shows the exact path to the logs. The feature enable logs are found in the WebSphere Commerce instances logs. The slide shows the exact path to the logs.



## Reference application failure symptoms

- No starter store listed under publisher
  - ▶ Verify sar files copied to correct directory
- SystemOut.log shows missing classes
  - ▶ Verify component services enabled by checking log
- Starter store entry page has blank background
  - ▶ Verify dojo041 directory present directly under stores.war
- Starter store partially functional, but missing new user interface features
  - ▶ Ensure browser has JavaScript enabled

Several failure symptoms are listed with steps to perform additional problem determination.

If no starter store is listed on the publish page, then verify the store archives are in the proper directory.

If the SystemOut.log shows missing classes, verify component services has been enabled by checking the enable logs under the instances directory.

If the starter store entry page has a blank background , verify the dojo041 directory exists directly under stores.war. If the dojo041 directory is not present, then either the intermediate dojo041.zip file was not constructed correctly, or that the file was not applied correctly when updating with the administration console.

If the starter store entry page is partially functional, but is missing the new user interface features, verify that the browser has JavaScript enabled.

## Summary

- WebSphere Commerce Web 2.0 feature is an implementation of a rich user interface typified by the Web 2.0 evolution
- AJAX and Dojo toolkit technologies play a large part in enabling rich user interfaces



This presentation showed that the WebSphere Commerce Web 2.0 feature is an implementation of a rich user interface typified by the Web 2.0 evolution. AJAX and Dojo toolkit technologies play a large part in enabling rich user interfaces.

## References

- What is Web 2.0 by Tim O'Reilly  
<http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>
- AJAX – Getting started  
[http://developer.mozilla.org/en/docs/AJAX:Getting\\_Started](http://developer.mozilla.org/en/docs/AJAX:Getting_Started)
- Dojo Developer Guide  
<http://dojotoolkit.org/docs/book>



For more information regarding Web 2.0, AJAX and Dojo, visit the following sites.

## 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  
WCSv602 Web20 Store Model Overview.ppt](mailto:iea@us.ibm.com?subject=Feedback%20about%20WCSv602%20Web20%20Store%20Model%20Overview.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:

IBM            WebSphere

Internet Explorer, Microsoft, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

JavaScript, 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 NONINFRINGEMENT. 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.